Sublinear-Time Clustering Oracle for Signed Graphs

Stefan Neumann [1] **Pan Peng** [2]

Abstract

Social networks are often modeled using *signed* graphs, where vertices correspond to users and edges have a sign that indicates whether an interaction between users was positive or negative. The arising signed graphs typically contain a clear community structure in the sense that the graph can be partitioned into a small number of *polarized* communities, each defining a sparse cut and indivisible into smaller polarized sub-communities. We provide a *local clustering oracle* for signed graphs with such a clear community structure, that can answer membership queries, i.e., "*Given a vertex v, which community does v belong to?*", in sublinear time by reading only a small portion of the graph. Formally, when the graph has bounded maximum degree and the number of communities is at most $O(\log n)$, then with $\tilde{O}(\sqrt{n}\,\mathrm{poly}(1/\epsilon))$ preprocessing time, our oracle can answer each membership query in $\tilde{O}(\sqrt{n}\,\mathrm{poly}(1/\epsilon))$ time, and it correctly classifies a $(1-\epsilon)$-fraction of vertices w.r.t. a set of hidden planted ground-truth communities. Our oracle is desirable in applications where the clustering information is needed for only a small number of vertices. Previously, such local clustering oracles were only known for *un*signed graphs; our generalization to signed graphs requires a number of new ideas and gives a novel spectral analysis of the behavior of *random walks with signs*. We evaluate our algorithm for constructing such an oracle and answering membership queries on both synthetic and real-world datasets, validating its performance in practice.

[1]KTH Royal Institute of Technology, Stockholm, Sweden [2]School of Computer Science and Technology, University of Science and Technology of China, Hefei, China. Correspondence to: Pan Peng <ppeng@ustc.edu.cn>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

1. Introduction

Finding clusters (or communities) in graphs is a well-studied and fundamental problem in computer science. While classically this problem has been studied in unsigned graphs, several recent works have focused on *signed* graphs, where each edge has a sign indicating whether the interaction between two nodes was friendly or hostile. This setting has been motivated by polarization in social networks, where the users form groups that have mostly friendly interactions within each group but there may exist hostile interactions between opposing groups (see, e.g., (Bonchi et al., 2019; Xiao et al., 2020; Ordozgoiti et al., 2020; Atay & Liu, 2020)).

More concretely, in a signed graph $G = (V, E, \sigma)$, each edge $e = (u, v) \in E$ is associated with a sign $\sigma(e) \in \{+, -\}$ indicating a positive (friendly) or negative (hostile) relation between the two vertices u and v. To model polarization, Harary (1953) proposed the notion of balanced graphs: a graph is *balanced* if it can be partitioned into two subsets V_1 and V_2 such that the induced subgraphs $G[V_1]$ and $G[V_2]$ only contain positive edges, while all edges with one endpoint in V_1 and the other endpoint in V_2 have a negative sign. For example, in a social network like Twitter the groups V_1 and V_2 could correspond to users of opposing opinions (e.g., Democrats and Republicans) that have a conflict but that behave nicely within their respective groups.

To detect polarization in social networks, several recent works have aimed at finding *nearly-balanced* communities inside signed graphs, i.e., their goal was to find induced subgraphs that after the removal of only few edges become balanced and are sparsely connected to the outside (Bonchi et al., 2019; Mercado et al., 2019; Xiao et al., 2020). Often the resulting communities are *minimal* in the sense that they are nearly-balanced and they cannot be further divided into smaller nearly-balanced communities. We will also refer to these communities as *polarized*. The main drawback of many existing methods for finding polarized communities is that they are inherently *global*, i.e., they need to process the full graph and return a partitioning of *all* vertices. In practice, however, the graphs are often so large that methods which aim to cluster all vertices have prohibitively high running times. Additionally, when mining social networks, the full graph is often not available because social network providers, such as Twitter, limit the amount of data that is available due to privacy constraints.

Fortunately, in many settings we only require the community membership information *for a small number of vertices*, or we just want to know if two given vertices belong to the same cluster or not. This could be the case, for example, when an analyst wishes to find out whether two users are part of the same polarized discussion or not. Furthermore, even in settings when the amount of data is limited (like in the Twitter example above), it seems feasible to explore the local neighborhood (e.g., by performing random walks) of each user that shall be classified.

Our contributions. We provide a *local clustering oracle for signed graphs*. The oracle preprocesses a small part of the graph and after the preprocessing finished, for a queried vertex v it can answer the following query:

- WHICHCLUSTER(v): Returns which cluster v belongs to.

Here, we assume that there is a set of (hidden) ground-truth clusters and WHICHCLUSTER(v) returns the index of the cluster v belongs to. Ideally, when two nodes u, v belong to the same ground-truth cluster, then the queries WHICHCLUSTER(u) and WHICHCLUSTER(v) will return the same result, and if u, v belong to different clusters, the results will be different.

Both the preprocessing time as well as the query time of our clustering oracle are *sublinear* in the size of the input graph. This is particularly useful when the clustering information is only required for a small number of vertices. More concretely, our oracle has preprocessing time[1] $\tilde{O}(\sqrt{n}\,\mathrm{poly}(1/\epsilon))$, where n is the number of vertices in the graph and $\epsilon > 0$ is an error parameter. The query procedure has query time $\tilde{O}(\sqrt{n}\,\mathrm{poly}(1/\epsilon))$. Such clustering oracles have been previously studied for unsigned graphs (Peng, 2020) from a theoretical point of view but none were known for signed graphs and it was unclear how well they perform in practice.

In a nutshell, the query procedure WHICHCLUSTER(v) performs $\tilde{O}(\sqrt{n}\,\mathrm{poly}(1/\epsilon))$ random walks of length $O(\log n)$ starting at v and then aggregates the information from these random walks into a sparse vector $\mathbf{m}_v$ with $\tilde{O}(\sqrt{n}\,\mathrm{poly}(1/\epsilon))$ non-zero entries. We define a pseudo-metric Δ on the space of these vectors and show that the vectors of vertices from the same community have "small" Δ-distance, while the vectors of vertices from different communities have "large" Δ-distance. We describe the details in Sec. 3.1.

Example 1. *One possible application of our oracle is the clustering of an online social networks such as Twitter. The user interactions on Twitter can be interpreted as a signed graph with some hidden communities. Now it is possible to label (e.g., by hand) a small number of users for several nearly-balanced communities. These users can be used as seed nodes for the oracle and then the oracle can be used to efficiently classify users based on which community they belong to. This addresses the issue that the Twitter graph is too large to cluster it completely. Additionally, as researchers we do not have access to the full Twitter graph but it seems feasible to perform a small number of short random walks from each user that shall be classified.*

We provide a theoretical analysis of the oracle for bounded-degree graphs with a constant (or logarithmic) number of "well-behaved" nearly-balanced communities. We show that when we apply WHICHCLUSTER(v) to all vertices, then the oracle classifies a $(1-\epsilon)$-fraction of the vertices correctly. See Thm. 4 for the formal statement of our result. To obtain this result, we relate the spectrum of the graph's normalized signed Laplacian to the random walks performed by the query procedures. Therefore, we give a novel spectral analysis of the behavior of *random walks with signs*, which essentially keep track whether the random walk traversed an even or an odd number of negative edges. Then we relate the Δ-distance of signed random walk vectors to the eigenvalues/eigenvectors of the signed Laplacian. See Sec. 3.2 for a precise statement of our technical contribution and App. D for an overview of our analysis.

Our theoretical contributions. Our theoretical results are inspired by sublinear clustering oracles for *un*signed graphs, and some notions and lemmas look superficially similar to the counterparts in unsigned graphs (e.g., (Czumaj et al., 2015; Peng, 2020)). However, to generalize these oracles to signed graphs we need several new ideas. We will now briefly discuss these new ideas.

First, the clustering oracle in unsigned graphs is based on the following intuitive idea: a random walk starting from a randomly chosen vertex of some cluster U will first be trapped in U, and the corresponding distribution converges to the uniform distribution on U (for simplicity, we assume the graph is regular); later, the random walk moves out of U and then the distribution converges to the uniform distribution on the whole graph. In signed graphs, this intuition is no longer true. In particular, the distribution of a signed random walk does not necessarily converge to a stationary distribution (if it exists). Interestingly, we show that (informally) in a polarized cluster U with a bipartition V_1 and V_2 corresponding to the two opposing groups, a signed random walk converges to either a scaled version of the uniform distribution on V_1, or a scaled version of the uniform distribution on V_2. To this end, we show that (see Lem. 6) if we map vertices to the spectral embedding defined by the first k eigenvectors of the signed normalized Laplacian matrix of the graph, then the embedded points are centered around *two opposite* centers. To contrast, in

[1] Here, $\tilde{O}(f(n))$ denotes running times $O(f(n) \cdot \mathrm{polylg}(n))$.

unsigned graphs, the spectral embedding of most vertices in the same cluster are close to one *single* center. To show the existence of two opposite centers, we develop a new property relating the eigenvectors and the polarized clusters, which may be useful for future work on clustering in signed graphs.

Second, for unsigned oracles, it suffices to consider the ℓ_2^2-distance between two random walk distributions starting from any two vertices u, v to decide if u, v are similar or not (i.e., if they belong to the same cluster or not). For signed oracles, since each polarized cluster has two opposite centers, we need to introduce a pseudometric distance $\tilde{\Delta}$ between the corresponding vectors to compare the similarity of two vertices. That is, for any two vertices u, v with random walk vectors $\mathbf{m}_u, \mathbf{m}_v$, we define

$$\tilde{\Delta}_{u,v} := \min\{\|\mathbf{m}_u - \mathbf{m}_v\|_2^2, \|\mathbf{m}_u + \mathbf{m}_v\|_2^2\}.$$

Intuitively, if u, v belong to the same polarized cluster with a bipartition V_1, V_2, then either the distance $\|\mathbf{m}_u - \mathbf{m}_v\|_2^2$ is small (corresponding to the case that u, v belong to the same part in the bipartition), or $\|\mathbf{m}_u + \mathbf{m}_v\|_2^2$ is small (corresponding to the case that u, v belong to two different parts). Furthermore, if u, v belong to two different clusters, then neither of these two distances is small.

Third, to characterize the cluster structure of a signed graph G, it is somehow natural to use the *signed bipartiteness ratio* (see Sec. 2), a signed analogue of conductance in unsigned graphs. However, we find that one *cannot* use the signed bipartiteness ratio of a graph to characterize the *inside* structure of a potential polarized cluster (see App. B). We resolve this by introducing a new notion called *inner signed bipartiteness ratio* $\beta^{\text{inner}}(G)$ of G that is a minimization function by considering all vertex subsets of at most half the total volume of the graph (see Eqn. (1) and Def. 3).

Our practical contributions. We provide the first implementations of signed *and unsigned* oracles. While our signed oracles with theoretical guarantees can only distinguish between different communities, we also provide a heuristic extension which allows for queries of the type: "In which opposing group of a community is vertex v?" In practice, such a query could be used, e.g., to decide whether a user in a social network is a Democrat or a Republican.

We evaluate our algorithms on synthetic and on real-world datasets (Sec. 4) and show that our oracles are practical. Our methods work well *even when the graphs do not satisfy the bounded degree assumption from our theoretical analysis*. Furthermore, our algorithms outperform existing methods when the graphs contain large communities. We further provide novel real-world datasets which contain signed graphs with a small number of large ground-truth communities; to the best of our knowledge, these are the first public datasets with this property and we make them freely available.

Related work. Finding communities in signed graphs has received a lot of attention. One line of works models polarized communities as (nearly) balanced subgraphs in a signed graph (Kunegis et al., 2010; Chiang et al., 2012; Bonchi et al., 2019; Cucuringu et al., 2019; 2020; Mercado et al., 2019; Xiao et al., 2020; Chu et al., 2016; Chiang et al., 2014; 2012). Xiao et al. (2020) provide a local algorithm for finding nearly balanced subgraphs. The algorithm of (Xiao et al., 2020) requires a set of seed nodes and returns a subgraph with small signed bipartiteness ratio; we compare our algorithm against this work in the experiments. Ordozgoiti et al. (2020) find large (exactly) balanced subgraphs. In another line of work, polarized communities were modeled using k-way balanced graphs (Chiang et al., 2012) and signed stochastic block models (Mercado et al., 2016; 2019) or using correlation clustering (Bansal et al., 2004; Cesa-Bianchi et al., 2012); these results are not directly comparable to our work since we consider k disjoint (nearly) 2-way balanced subgraphs while these works try to find a single partitioning of the graph that reveals k communities. Interestingly, many of these works are based on spectral graph theory (e.g., (Kunegis et al., 2010; Chiang et al., 2012; Xiao et al., 2020; Ordozgoiti et al., 2020; Mercado et al., 2016; 2019)).

Jung et al. (2016; 2020) use signed random walks with restarts to rank users in social networks but, unlike in our work, they do not relate the signed random walks to the spectrum of the signed graph.

Sublinear-time algorithms for clustering *unsigned* graphs have been studied using the notion of conductance (rather than the signed bipartiteness ratio). Czumaj et al. (2015) gave a property testing algorithm which can decide whether a graph is k-clusterable or is *far from* being k-clusterable in sublinear time. Interestingly, the algorithm by Czumaj et al. (2015) can be adapted to a sublinear-time clustering oracle. Peng (2020) extended this to a robust clustering oracle that reports the clustering information of graphs with noisy partial information. Chiplunkar et al. (2018) and Gluch et al. (2021) provided further improvements.

Intriguingly, the algorithm by Pons & Latapy (2006) is also based on clustering the vectors of short random walks and is very popular in practice (e.g., it is implemented in the *igraph* software package (Csardi & Nepusz, 2006)). The similarity measure used in Pons & Latapy (2006) is quite similar to the one which was independently proposed by Czumaj et al. (2015), though the latter is focusing on using a small number of random walks to estimate the measure (rather than computing it directly) and thus achieving a sublinear-time algorithm. Therefore, one can view the results of Czumaj et al. (2015) and of this paper as a further theoretical justification for the practical success of the work of Pons & Latapy (2006).

2. Preliminaries

Let $G = (V, E, \sigma)$ be an unweighted signed graph with n vertices, m edges and edge signs $\sigma(e) \in \{+, -\}$. The degree $d_G(u)$ of a vertex u is $d_G(u) = |\{v : (u,v) \in E\}|$; note that the degree does not take into account the signs of the edges. For any set $S \subseteq V$, let $\mathrm{vol}_G(S) = \sum_{u \in S} d_G(v)$. The volume of G is $\mathrm{vol}(G) = \sum_{u \in V} d_G(u)$.

For $V_1, V_2 \subseteq V$, we set $E(V_1, V_2) = \{(u,v) \in E : u \in V_1, v \in V_2\}$. Furthermore, we set $E^+(V_1, V_2) = \{(u,v) \in E(V_1, V_2) : \sigma(uv) = +\}$ and $E^-(V_1, V_2) = \{(u,v) \in E(V_1, V_2) : \sigma(uv) = -\}$. When $V_1 = V_2$, we set $E(V_1) = E(V_1, V_1)$. To maintain consistency with previous works, we set $|E(V_1)|$ to twice the number of edges in $G[V_1]$ (but we do not make this change for $|E(V_1, V_1)|$). We define $|E^+(V_1)|$ and $|E^-(V_1)|$ analogously to $|E(V_1)|$.

Signed bipartiteness ratio. Following the work of Xiao et al. (2020), we use the *signed bipartiteness ratio* to capture polarization between two opposing groups in a graph. A pair (V_1, V_2) is a *sub-bipartition* of V if $\emptyset \neq V_1 \cup V_2 \subseteq V$ and $V_1 \cap V_2 = \emptyset$. For a sub-bipartition (V_1, V_2) of V we set $\mathrm{vol}_G(V_1, V_2) = \sum_{u \in V_1 \cup V_2} d_G(u)$. Now the *signed bipartiteness ratio* of (V_1, V_2) is given by

$$\beta_G(V_1, V_2) := \frac{e_G(V_1, V_2)}{\mathrm{vol}_G(V_1, V_2)},$$

where

$$\begin{aligned} e_G(V_1, V_2) = & 2\left|E_G^+(V_1, V_2)\right| + \left|E_G^-(V_1)\right| + \left|E_G^-(V_2)\right| \\ & + \left|E_G(V_1 \cup V_2, \overline{V_1 \cup V_2})\right|. \end{aligned}$$

Observe that when $\beta_G(V_1, V_2)$ is small then the induced subgraph $G[V_1 \cup V_2]$ is close to balanced (i.e., $G[V_1]$ and $G[V_2]$ contain only few negative edges and there are mostly negative edges between V_1 and V_2), *and* the vertices in $V_1 \cup V_2$ are sparsely connected to the rest of the graph (i.e., there are few edges from $V_1 \cup V_2$ to $V \setminus (V_1 \cup V_2)$).

For a set of vertices $\emptyset \neq U \subseteq V$, we define *the signed bipartiteness ratio of U in G* as $\beta_G(U) := \min_{(V_1, V_2):\, V_1 \cup V_2 = U} \beta_G(V_1, V_2)$, where the minimum is taken over all partitions (V_1, V_2) of U. For a graph G, we define the *(classic) signed bipartiteness ratio* of G as

$$\begin{aligned} \beta(G) &:= \min_{\emptyset \neq U \subseteq V} \beta_G(U) \\ &= \min_{(V_1, V_2):\text{sub-bipartition of } V} \beta_G(V_1, V_2). \end{aligned}$$

Observe that a set of vertices U is balanced *if and only if* U can be partitioned into subsets V_1 and V_2 such that $\beta_{G[U]}(V_1, V_2) = 0$ *if and only if* $\beta(G[U]) = 0$. Thus, one can interpret the signed bipartiteness ratio as a measure for how close a certain subgraph is to being balanced. The sets V_1 and V_2 that partition U are sometimes called *biclusters*.

Spectral signed graph theory. Next, we introduce definitions for the spectral analysis of signed graphs. We use bold letters to denote vectors and matrices. Let $\mathbf{D}$ be the $n \times n$ diagonal degree matrix of G, i.e., $\mathbf{D}_{uu} = d_G(u)$ for all $u \in V$. Let $\mathbf{A}^\sigma$ be the $n \times n$ signed adjacency matrix, i.e., $\mathbf{A}^\sigma_{uv} = \sigma(uv)$ if $(u,v) \in E$, and $\mathbf{A}^\sigma_{uv} = 0$, otherwise. Let $\mathbf{I}$ be the $n \times n$ identity matrix.

We call $\mathbf{L}^\sigma := \mathbf{D} - \mathbf{A}^\sigma$ the *signed (unnormalized) Laplacian* matrix, and $\mathcal{L}^\sigma := \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{A}^\sigma\mathbf{D}^{-1/2}$ the *signed normalized Laplacian* matrix. It is well-known that all eigenvalues of $\mathcal{L}^\sigma$ are in the interval $[0, 2]$ and we list them in non-decreasing order as $0 \leq \lambda_1 \leq \cdots \leq \lambda_n \leq 2$.

For $k \in [n]$, the *k-way signed bipartiteness ratio of G* is

$$\begin{aligned} \beta_k(G) &:= \min_{U_1, \ldots, U_k} \max_{i=1,\ldots,k} \beta_G(U_i) \\ &= \min_{\{(V_{2i-1}, V_{2i})\}_{i=1}^k} \max_{i=1,\ldots,k} \beta_G(V_{2i-1}, V_{2i}). \end{aligned}$$

Here, the minima are taken over all possible choices of k non-empty, disjoint sets $U_i \subset V$ and disjoint sub-bipartitions (V_{2i-1}, V_{2i}), respectively. Intuitively, $\beta_k(G)$ is small iff G contains k disjoint communities that are close to balanced iff G contains k polarized communities.

Atay & Liu (2020) provided a Cheeger-type inequality that relates the k-way signed bipartiteness ratio to the eigenvalues λ_k of the signed normalized Laplacian $\mathcal{L}^\sigma$.

Theorem 2 (Higher-order Signed Cheeger Inequality (Atay & Liu, 2020))**.** *There exists a constant C_1 such that for all signed graphs G and $k \in [n]$, $\frac{\lambda_k}{2} \leq \beta_k(G) \leq C_1 k^3 \sqrt{\lambda_k}$.*

Finally, $\mathbf{W} = \frac{\mathbf{I} + \mathbf{D}^{-1}\mathbf{A}^\sigma}{2}$ is the *walk matrix* that corresponds to lazy random walks in a signed graph G. Additionally, for $t \in \mathbb{N}$ and $v \in V$, we set $\mathbf{p}_v^t = \mathbf{1}_v \mathbf{W}^t$, where $\mathbf{1}_v$ is the n-dimensional indicator vector that has a 1 in the v'th entry and is 0 in all other entries.

3. Main Result and Algorithm

In this section, we formally present our main result and give the details of our clustering oracle. To state our theorem, we first need to introduce two new definitions. For a signed graph $G = (V, E, \sigma)$, we define the *inner signed bipartiteness ratio* of G as

$$\beta^{\text{inner}}(G) := \min_{\emptyset \neq U \subseteq V:\ \mathrm{vol}(U) \leq \frac{1}{2}\mathrm{vol}(G)} \beta_G(U). \quad (1)$$

Note that we only consider subsets U *of volume at most* $\frac{1}{2}\mathrm{vol}(G)$; this is in contrast with the definition of $\beta(G)$, in which the minimum is taken over *all possible* subsets U. The definition (1) resembles the inner conductance that has been used to study the clusterability of unsigned graphs (e.g., (Gharan & Trevisan, 2014; Czumaj et al., 2015)), though in contrast with $\beta(G)$, it is *not* directly associated with the signed Cheeger inequality.

Next, we define the notion of clusterability under which we will obtain our theoretical results.

Definition 3. *Let $k \in \mathbb{N}$, $\beta_{in}, \beta_{out} \in \mathbb{R}_{>0}$ and let $G = (V, E, \sigma)$ be an unweighted signed graph. We say that G is* $(k, \beta_{\text{in}}, \beta_{\text{out}})$-clusterable *if there exists a partition of V into k disjoint subsets $(U_1, \ldots, U_k)$ such that $\beta_G(U_i) \le \beta_{out}$ and $\beta^{inner}(G[U_i]) \ge \beta_{in}$ for all $i \in [k]$. Each subset U_i is called a* $(\beta_{\text{in}}, \beta_{\text{out}})$-cluster *and the corresponding partitioning is called a* $(k, \beta_{\text{in}}, \beta_{\text{out}})$-clustering*. Furthermore, if each subset U_i satisfies that $|U_i| \ge \Omega(\frac{n}{k})$, then we call the partition $(U_1, \ldots, U_k)$ a* balanced *$(k, \beta_{in}, \beta_{out})$-clustering.*

Let us briefly explain this definition; it is handy to think of β_{out} as very small and $\beta_{\text{out}} \ll \beta_{\text{in}}$. The first condition that $\beta_G(U_i) \le \beta_{\text{out}}$ for all $i = 1, \ldots, k$ ensures that the graph contains k communities which are nearly-balanced and that can be viewed as polarized communities. The second condition ($\beta^{\text{inner}}(G[U_i]) \ge \beta_{\text{in}}$ for all $i = 1, \ldots, k$) ensures that each of the nearly-balanced communities is minimal in the sense that it cannot be further decomposed into more balanced communities.

We remark that at first glance it might be surprising that in Definition 3, we use $\beta^{\text{inner}}(G[U_i])$ instead of $\beta(G[U_i])$ to measure the indivisibility of $G[U_i]$ into smaller nearly-balanced communities. However, in App. B we show that if $\beta_G(U_i)$ is small, then $\beta(G[U_i])$ is also small. This indicates that $\beta(G[U])$ is not an appropriate measure for this characterization.

Now we state our main result. We consider signed graphs G of degree at most d, where d is a constant throughout the paper. We assume that we have query access to the adjacency list of G, i.e., for any vertex v and an index $i \le d$, we can query the i-th neighbor of v in constant time if it exists (if no such neighbor exists we get a special symbol '$\perp$'). Let $P \triangle Q$ denote the symmetric difference. In the following, we let $d > 10$, $k \ge 1$, $\epsilon \in (0, 1)$ and $\beta_{\text{in}} \in (0, 1)$. Let n be an integer such that $n \ge \frac{1800k^2 \log(k)}{\gamma\epsilon}$, where $\gamma \in (0, 1]$ is a constant.

Theorem 4. *Let $G = (V, E, \sigma)$ be a signed graph with $|V| = n$ vertices and maximum degree at most d. Suppose that G has a balanced $(k, \beta_{in}, \beta_{out})$-clustering $U_1, \cdots, U_k$, $\beta_{out} < \frac{\epsilon\beta_{in}^2}{C' \log(k) k^7 d^3 \log n}$, where C' is some sufficiently large constant, and $|U_i| \ge \gamma\frac{n}{k}$ for all $i = 1, \ldots, k$. There exists an algorithm that has query access to the adjacency list of G and constructs a clustering oracle in $O(\sqrt{n} \cdot \text{poly}(\frac{kd \cdot \log n}{\epsilon\beta_{in}}))$ preprocessing time. Furthermore, with probability at least 0.9, the following hold:*

1. *Using the oracle, the algorithm can answer any* WHICHCLUSTER *query in $O(\sqrt{n} \cdot \text{poly}(\frac{kd \cdot \log n}{\epsilon\beta_{in}}))$ time.*
2. *Let $P_i := \{u \in V : \text{WHICHCLUSTER}(u) = i\}$, $i \in [k]$, be the clusters defined by* WHICHCLUSTER*. Then there exists a permutation $\pi : [k] \to [k]$ such that for all $i \in [k]$, $\left|P_{\pi(i)} \triangle U_i\right| \le O(\epsilon/\log k)|U_i|$.*

The theorem asserts that if the input graph G has bounded degree and satisfies the assumptions from Def. 3 with $\beta_{\text{out}} \ll \beta_{\text{in}}$, then our clustering oracle has preprocessing and query time $\tilde{O}(\sqrt{n}\,\text{poly}(1/\epsilon))$. Furthermore, the second item implies that if we pick ϵ small enough, we can make the number of misclassified vertices arbitrarily small. In particular, for any $\delta > 0$ we can pick ϵ such that the oracle classifies at least a $(1 - \delta)$-fraction of the vertices correctly.

3.1. The Algorithm

Now we present the implementation of our clustering oracle. The main building block of our oracle are lazy signed random walks which we will discuss first. Based on a sequence of random walks starting at a vertex u, we will define a sparse vector $\mathbf{m}_u$. We will then use the vectors $\mathbf{m}_u$ and $\mathbf{m}_v$ to estimate distances δ_{uv} between vertices u and v with the intuition that u and v are in the same cluster iff δ_{uv} is small. We will also discuss the preprocessing of the oracle and the query procedures.

Lazy signed random walks. We introduce lazy signed random walks. Intuitively, a lazy signed random walk is a lazy random walk on the unsigned version of the graph that keeps track of the sign of the walk. Here, the sign of the walk is the multiplication of the signs of all traversed edges.

More formally, a *lazy signed random walk* of length t from a vertex u proceeds as follows. Initially, at step $T = 0$, we start at vertex $u_0 := u$ with sign $s_0 := +$. Suppose that at step $0 \le T < t$ we are at vertex u_T with sign $s_T \in \{+, -\}$. Then at the step $T + 1$, with probability $\frac{1}{2}$ we stay at u_T and keep the sign unchanged (i.e., $u_{T+1} = v_t$, $s_{T+1} = s_T$), and with the remaining $\frac{1}{2}$ probability, we choose a random neighbor v of u_T with probability $\frac{1}{d_G(u_T)}$, and move to v and update $u_{T+1} = v$, $s_{T+1} = \sigma(e_{T+1})s_T$, where $e_{T+1} = \{u_T, v\}$. Thus, if a walk traverses the edges $e_1, \ldots, e_t$ then the final sign of the walk is $\prod_{i=1}^{t} \sigma(e_i)$. Later, we will set the number of steps to $t = \Theta(\log n)$.

Vectors from sequences of random walks. Next, given a start vertex u, we describe how to obtain a sparse vector $\mathbf{m}_u \in \mathbb{R}^n$ based on a sequence of lazy signed random walks. In Sec. D, we will argue that $\mathbf{m}_u$ essentially serves as a (sparse) approximation of the vector $\mathbf{p}_u^t \mathbf{D}^{-1/2}$, where $\mathbf{p}_u^t = \mathbf{1}_u \mathbf{W}^t$, $\mathbf{W}$ is the walk matrix and $\mathbf{D}$ is the degree matrix as defined in Sec. 2.

Suppose that we perform R lazy signed random walks of length t from vertex u and let $v_1, \ldots, v_R$ be the vertices at which these random walks finish with respective signs

$s_1, \ldots, s_R$. Now we define two vectors $\mathbf{m}_u^+, \mathbf{m}_u^- \in \mathbb{R}_{\geq 0}^n$ as follows. We set $\mathbf{m}_u^+(v)$ to the fraction of random walks that ended in vertex v with sign $+$, i.e., $\mathbf{m}_u^+(v) = \frac{|\{i\colon v_i = v, s_i = +\}|}{R}$ for all $v \in V$. Similarly, we set $\mathbf{m}_u^-(v) = \frac{|\{i\colon v_i = v, s_i = -\}|}{R}$. Finally, we set $\mathbf{m}_u = (\mathbf{m}_u^+ - \mathbf{m}_u^-)\mathbf{D}^{-1/2}$.

Note that $\mathbf{m}_u$ can have positive *and negative* entries. Furthermore, the R random walks can end in at most R different vertices, and thus $\mathbf{m}_u$ has at most R non-zero entries. Later, we will set $R = \tilde{O}(\sqrt{n}\,\mathrm{poly}(1/\epsilon))$.

We now introduce our key subroutine ESTDOTPROD(u,v,t,α) and the pseudocode of the routine is presented in Alg. 1. Consider two vertices u and v, the random walk length t and a technical parameter α that we will set in the proof of Thm. 4. Then ESTDOTPROD(u,v,t,α) computes the two vectors $\mathbf{m}_u$ and $\mathbf{m}_v$ and calculates their dot product. This is repeated $h = O(\log n)$ times and then the median of these dot products is returned. We will later (Lem. 12) show that the output of ESTDOTPROD(u,v,t,α) gives an approximation of $\langle \mathbf{p}_u^t \mathbf{D}^{-1/2}, \mathbf{p}_v^t \mathbf{D}^{-1/2} \rangle$ with small error.

Preprocessing. We present the preprocessing phase of the oracle in Alg. 2. Here, we make use of the fact (see Sec. D for details) that when two vertices u and v are from the same cluster, then

$$\Delta_{uv} := \min\{\|\mathbf{p}_u^t \mathbf{D}^{-1/2} - \mathbf{p}_v^t \mathbf{D}^{-1/2}\|_2^2, \|\mathbf{p}_u^t \mathbf{D}^{-1/2} + \mathbf{p}_v^t \mathbf{D}^{-1/2}\|_2^2\}$$

should be small, whereas if u and v are from different clusters then Δ_{uv} should be large. However, since our algorithm has no access to the vectors $\mathbf{p}_u^t$ and $\mathbf{p}_v^t$, we will need to use ESTDOTPROD to obtain an estimate δ_{uv} as approximation of Δ_{uv}.

The preprocessing starts by sampling a set S of $O(k \log k)$ vertices. Essentially this ensures that from each of the k clusters, S contains at least one vertex. Now for all pairs of vertices $u, v \in S$, we compute δ_{uv} as approximation of Δ_{uv} by rewriting the norms inside the definition Δ_{uv} as dot products and then estimating each of these dot products using ESTDOTPROD. More concretely, we observe that

$$\Delta_{uv} = \min\{Y_{uu} + Y_{vv} - 2Y_{uv}, Y_{uu} + Y_{vv} + 2Y_{uv}\}, \quad (2)$$

where $Y_{ab} = \langle \mathbf{p}_a^t \mathbf{D}^{-1/2}, \mathbf{p}_b^t \mathbf{D}^{-1/2} \rangle$ for $a, b \in V$. Next, we let $X_{ab} = \mathrm{ESTDOTPROD}(a, b)$ and we define

$$\delta_{uv} = \min\{X_{uu} + X_{vv} - 2X_{uv}, X_{uu} + X_{vv} + 2X_{uv}\}. \quad (3)$$

We cluster the vertices in S as follows. We create an auxiliary graph H with vertex set S and without edges. Then we add edges for each pair of vertices u and v such that δ_{uv} is "small", i.e., if $\delta_{uv} < \frac{1}{2dn}$. In our proof of Thm. 4, we will show that if the conditions of the theorem hold, then H consists of k cliques corresponding to the k clusters in the $(k, \beta_{\text{in}}, \beta_{\text{out}})$-clustering. Thus, the preprocessing will correctly identify at least one vertex from each cluster.

Query procedure. For a query WHICHCLUSTER(v), we proceed similarly to how we clustered the vertices in S in the preprocessing. More concretely, given v as input to the query, we compute δ_{uv} for all $u \in S$. If $\delta_{uv} \leq \frac{1}{2dn}$ for some $u \in S$ then we return that v belongs to the cluster of u. For the full details, see Alg. 3.

3.2. Main Technical Contribution

To prove Thm. 4, our global proof strategy is as follows. First, we show that two vertices u and v are from the same cluster *if and only if* Δ_{uv} is small (see Lem.s 9 and 10). Second, we show that X_{uv} does not introduce too much error for estimating $\langle \mathbf{p}_v^t \mathbf{D}^{-1/2}, \mathbf{p}_u^t \mathbf{D}^{-1/2} \rangle$ (see Lem. 12). This then implies that with a large probability δ_{uv} is close to Δ_{uv} and, thus, δ_{uv} is small *iff* u and v are from the same cluster. We give more intuition and details of our proof strategy in App. D.

This strategy is similar to the one by Czumaj et al. (2015) for unsigned graphs. However, even though our global proof strategy is similar, we still have to contribute a significant amount of new ideas to extend the clustering oracle from the unsigned to the signed setting. We will now discuss some of these challenges in more detail.

One particular challenge was showing that Δ_{uv} is small if u and v are from the same cluster (see Lem. 9). To prove this, we need two technical lemmas that constitute our main technical contribution. One of them (Lem. 6) provides a connection between bicluster membership and the entries of the eigenvectors of the signed normalized Laplacian. We believe this result is of independent interest and will find further applications in the analysis of signed graphs.

Let $\mathbf{v}_1, \cdots, \mathbf{v}_n$ be the orthonormal row eigenvectors of $\mathcal{L}^\sigma$ s.t. $\lambda_i \mathbf{v}_i = \mathbf{v}_i \mathcal{L}^\sigma$. Thus $\mathbf{v}_i \mathbf{v}_j^\top = 1$ if $i = j$ and $\mathbf{v}_i \mathbf{v}_j^\top = 0$ otherwise. Let $\mathbf{v}_i' = \mathbf{v}_i \mathbf{D}^{-1/2}$. For a subset $U \subseteq V$, let $\mu_U = \mathrm{vol}_{G[U]}(U)$ be the total volume of the induced graph $G[U]$, i.e., the sum of degrees of all vertices in $G[U]$.

The first lemma says there is a gap between λ_k and λ_{k+1} if a graph is $(k, \beta_{\text{in}}, \beta_{\text{out}})$-clusterable, which allows us to use the first k eigenvectors $\mathbf{v}_1, \ldots, \mathbf{v}_k$ to bound Δ_{uv}. The proof makes use of our new definition of inner signed bipartiteness ratio of a graph. We defer the proof details to App. E.1.

Lemma 5. *If G is signed and $(k, \beta_{in}, \beta_{out})$-clusterable, then $\lambda_i \leq 2\beta_{out}$ for all $i \leq k$ and $\lambda_i \geq \frac{\beta_{in}^2}{C_1^2 (k+1)^6}$ for all $i \geq k+1$, where C_1 is the constant from Thm. 2.*

For our second lemma consider a cluster U with

$\beta^{\text{inner}}(G[U]) \geq \beta_{\text{in}}$ and $\beta_G(U) \leq \beta_{\text{out}}$. Then there exists a partition of U into biclusters V_1 and V_2 such that $\beta_G(V_1, V_2) = \beta_G(U) \leq \beta_{\text{out}}$, i.e., V_1 and V_2 correspond to the two polarized groups inside cluster U. Intuitively, our lemma asserts that for most vertices $u \in U$, the sign of the entry $\mathbf{v}'_i(u)$ ($i = 1, \dots, k$) reveals whether $u \in V_1$ or $u \in V_2$. In other words, we show that each vector $\mathbf{v}'_i$ ($i = 1, \dots, k$), approximately reveals a biclustering of U into two polarized communities.

Slightly more precisely, we will show that if $i \leq k$ then for each "typical" vertex pair $u, v \in U$, it holds that $\mathbf{v}'_i(u) \approx \mathbf{v}'_i(v)$ if $u, v \in V_1$ or $u, v \in V_2$, and $\mathbf{v}'_i(u) \approx -\mathbf{v}'_i(v)$ otherwise. To do so, we establish a novel connection between the structure of each balanced cluster and the geometric embedding from these k eigenvectors: we relate the signed indicator vector $\mathbf{1}_{V_1,V_2}$ to the first eigenvector $\mathbf{w}_1$ of (the normalized signed Laplacian of) the subgraph $G[U]$, and we use the variational characterization of the second eigenvector of $G[U]$ to analyze the $\mathbf{v}'_i$ restricted on U. Here, $\mathbf{1}_{V_1,V_2} \in \mathbb{R}^U$ is the vector with $\mathbf{1}_{V_1,V_2}(u) = 1$ if $u \in V_1$ and $\mathbf{1}_{V_1,V_2}(u) = -1$ if $u \in V_2$. We present the proof of the lemma in App. E.2.

Lemma 6. *Let $\alpha \in (0, 1)$. Suppose $G = (V, E, \sigma)$ is signed and $(k, \beta_{in}, \beta_{out})$-clusterable. Let U be a cluster of G with $\beta^{inner}(G[U]) \geq \beta_{in}$ and $\beta_G(U) \leq \beta_{out}$. Then there exists a partition V_1, V_2 of U, a subset $\widetilde{U}$ of U, and constants $c_i, 1 \leq i \leq k$, such that $|c_i| \leq 3d$, $|\widetilde{U}| \geq (1 - \alpha)|U|$, and for each $i \leq k$,*

- *if $u \in V_1 \cap \widetilde{U}$, then $\left|\mathbf{v}'_i(u) - c_i \cdot \frac{1}{\sqrt{\mu_U}}\right| \leq \frac{64dC_1}{\beta_{in}} \cdot \sqrt{\frac{\beta_{out}}{\alpha \cdot \mu_U}}$*
- *if $u \in V_2 \cap \widetilde{U}$, then $\left|\mathbf{v}'_i(u) + c_i \cdot \frac{1}{\sqrt{\mu_U}}\right| \leq \frac{64dC_1}{\beta_{in}} \cdot \sqrt{\frac{\beta_{out}}{\alpha \cdot \mu_U}}$*

where C_1 is the constant from Thm. 2.

4. Experiments

We experimentally evaluated our algorithms on a MacBook Pro with 16 GB RAM and a 2 GHz Quad-Core Intel Core i5. Our algorithms were implemented in C++11.We always performed 8 WHICHCLUSTER-queries in parallel. See App. F for more implementation details. Our source code is available on github.[2]

Quality measure. In our evaluation, we consider a set of ground-truth clusters $U_1, \dots, U_k$ and the output of an algorithm $\tilde{U}_1, \dots, \tilde{U}_s$. We assume w.l.o.g. that $s \geq k$. The *accuracy* of the clustering is $\min_\pi \frac{1}{m} \sum_{i=1}^k \left|U_i \cap \tilde{U}_{\pi(i)}\right|$, where $\pi\colon [k] \to [s]$ is an injective function and $m = \sum_i |U_i|$ is the number of elements in the ground-truth clusters. Thus, the accuracy measures how many elements were classified correctly; since π is injective, each U_i must be mapped to a different $\tilde{U}_j$. When a cluster $\tilde{U}_j$ contains an element $v \notin \bigcup_i U_i$, we remove v from $\tilde{U}_j$ (this can happen when we do not have ground-truth information for all vertices).

Algorithms. First, we consider our signed oracles RW-SEEDED and RW-UNSEEDED, resp., obtain ground-truth seed nodes or randomly picked vertices in the preprocessing (see App. F). Second, we implemented two *un*signed oracles which operate on the underlying unsigned graph (see App. F); we denote them RW-U-SEEDED and RW-U-UNSEEDED, depending on their initialization.

As baselines we use FOCG by Chu et al. (2016) and POLARSEEDS by Xiao et al. (2020). FOCG is a *global* algorithm that requires access to the full graph and enumerates nearly-balanced communities. POLARSEEDS is a local algorithm that requires some seed nodes as input and explores the graph locally to find a subgraph with small signed bipartiteness ratio. We used the implementations provided by the authors and ran them with the default parameters.

Since our focus was on practically efficient oracle data structures, we did not compare against the oracle by Gluch et al. (2021), as it is of highly theoretical nature, and involves several subroutines that hinder the implementation in practice.[3]

We consider two types of experiments: (1) *Clustering* a graph G into polarized communities $U_1, \dots, U_k$ (as per Def. 3) and (2) *biclustering* G into opposing polarized groups $(V_1, V_2), (V_3, V_4), \dots, (V_{2k-1}, V_{2k})$. For the biclustering setting, we consider a heuristic of our oracle which does not take absolute values when computing $\mathbf{m}_x$ (see App. F for details). In both cases, we use the corresponding versions of the algorithms; to avoid blowing up the notations, we use the same algorithm names for the clustering and biclustering versions.

Evaluation on synthetic data. Due to lack of space, we present our experiments on synthetic data in App. G. Our experiments show that our oracles outperform the baselines when the clusters are large. The unsigned oracle works well for clustering but only the signed oracle can recover the biclusters (V_{2i-1}, V_{2i}). Also, the seeded oracles outperform the unseeded oracles and our oracle scale linearly in the number the number of steps and the walk lengths.

We also note that FOCG and POLARSEEDS do not perform very well on the synthetic datasets. We believe the main reason is that these algorithms were built to find "small" clusters which do not necessarily partition the graph; in contrast, our method is strongest in the presence of large clusters that partition the graph. This large-vs-small cluster intuition is also corroborated by our experiments on syn-

[2] `https://github.com/StefanResearch/signed-oracle`

[3] For instance, Alg. 10 in the arxiv version of Gluch et al. (2021) samples a set of $\Omega(k^4)$ vertices and then enumerates all possible partitions of this set. This would be infeasible in practice.

thetic data in Figure 3(b) in the appendix: as the number k of clusters increases, the clusters get smaller and the performance of POLARSEEDS improves. If we increased k further, the performance of POLARSEEDS would improve further and eventually outperform our methods.

Evaluation on real-world data. Since we are not aware of public signed graph datasets with a small number of large ground-truth communities, we created our own real-world data. We make them available on github.[2]

We obtained our graphs from English-language Wikipedia by considering Wikipedia pages about politicians and the articles linked on their pages. We selected five countries (UK, Germany, Austria, Spain and US) and for each we selected a number of politicians (UK: 2307, Germany: 1444, Austria: 190, Spain: 546, US: 5053); this gives the ground-truth clusters U_i. The politicians from the clusters U_i belong to one of two opposing parties, which splits each U_i into opposing groups V_{2i-1} and V_{2i}. In our graphs, the vertices correspond to Wikipedia pages of politicians and articles that are linked on the politicians' pages. An edge (u, v) indicates that page u has a link to page v. The sign for an edge (u, v) is $-$ if u and v are politicians from the same country and they are in opposing parties (e.g., Democrats and Republicans in the US); otherwise, we set the sign to $+$.

We created three dataset. WikiL contains all politicans and all articles linked on their pages; we included all edges that contain at least one politician. On WikiL, the signed bipartiteness ratios of the three large communities (UK, Germany, US) is ≈ 0.66 and for the smaller communities (Austria, Spain) it is ≈ 0.9. We also consider two smaller versions of WikiL: WikiS is the largest connected component of $G[P]$, where G is the graph given by WikiL and P is the set of politician nodes in WikiL; WikiM is the largest connected component of $G[P \cup V']$, where V' is a randomly sampled set containing 10% of the non-politician nodes from WikiL. Note that for WikiL and WikiM we only have a ground-truth clustering for a *subset* of the vertices (namely, for the politician nodes). We present statistics of the datasets in Table 1. In our experiments, we use the undirected versions of these datasets, i.e., each directed edge is made undirected.

Experiments on WikiS. Fig. 1 shows our results on WikiS. Unless stated otherwise, our oracles used 1000 random walks of length 20. For clustering (biclustering) experiments, the seeded oracles obtained 10 (5) seeds for each U_i (V_i); the unseeded oracles sampled $5k$ vertices, where $k = 5$ for clustering and $k = 10$ for biclustering. For the clustering experiments (Figs. 1(a) and 1(b)), the seeded oracles achieve almost perfect accuracy; the unseeded methods are worse and benefit from longer random walks (Fig. 1(a)). For the biclustering experiments (Figs. 1(c) and 1(d)), RW-SEEDED is by far the best method and achieves excellent accuracies. RW-U-SEEDED consistently achieves accuracies above 50% but below 60%; this suggests that RW-U-SEEDED successfully finds the clusters U_i (as shown by the clustering results) but, as it ignores the edge signs, it places the vertices into the biclusters V_{2i-1} and V_{2i} only slightly better than random. For biclustering, the unseeded methods do not perform well. FOCG and POLARSEEDS achieve low accuracies, since they return too small clusters.

Experiments on WikiM and WikiL. Fig. 2 presents our results on the larger datasets. We did not run the unseeded oracles, since in WikiM and WikiL not all vertices are contained in ground-truth communities. We used random walks of length 24 for WikiM and of length 20 for WikiL; we initialized the seeds as for WikiS. Fig. 2(a) shows that even on WikiL, the seeded oracles find the clusters U_i with almost perfect accuracy. Furthermore, the algorithms scale linearly in the number of random walks and on average queries takes less than 1.4 seconds (Fig. 2(b)). However, compared to WikiS, we obtain lower accuracies for the biclustering experiments: while on WikiM, RW-SEEDED still achieves accuracies over 83% with enough random walks (Fig. 2(c)), for WikiL even with 10 000 random walks, RW-SEEDED only achieves an accuracy of 50% (Fig. 2(d)). We blame this on the fact that in WikiL the ground-truth clusters are relatively small (they contain only 3.7% of the vertices). Additionally, in WikiL the fraction of negative edges is only 8%, and thus only few random walks will encounter a negative edge and RW-SEEDED cannot benefit from the edge-sign information. As before, FOCG and POLARSEEDS have low accuracies.

Conclusion. Our oracles are successful for finding polarized communities U_i, even when the graphs do not have bounded degrees (as in our theoretical analysis). RW-SEEDED is successful in finding the biclusters (V_{2i-1}, V_{2i}), as long as they are large enough and there are enough negative edges.

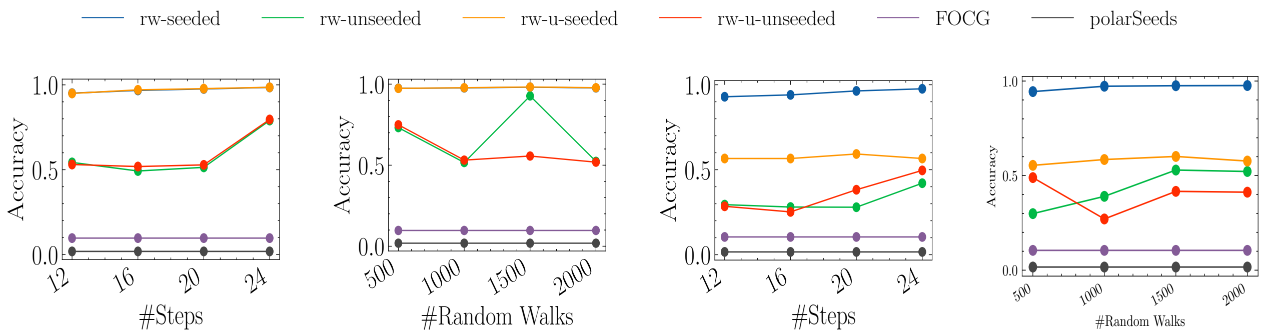
5. Conclusion

We presented a local clustering oracle for signed graphs. Given a vertex u, the oracle can return the cluster membership of u in sublinear time. Such a data structure is desirable when the input graphs are large and the cluster membership is only required for a small number of vertices. We proved that if the graph satisfies a clusterability assumption, then the oracle returns correct cluster memberships for a $(1 - \epsilon)$-fraction of the vertices w.r.t. a hidden set of ground-truth clusters. We also evaluated the oracle practically, showing that it achieves good results for large clusters.

In the future it will be interesting to provide a theoretical analysis for our biclustering heuristic; here, ideas from Trevisan (2012) might be helpful. From a practical point of view it would be interesting to obtain improvements for our biclustering heuristic, which allow to find small biclusters (V_{2i-1}, V_{2i}) when there are few negative edges. Two direc-

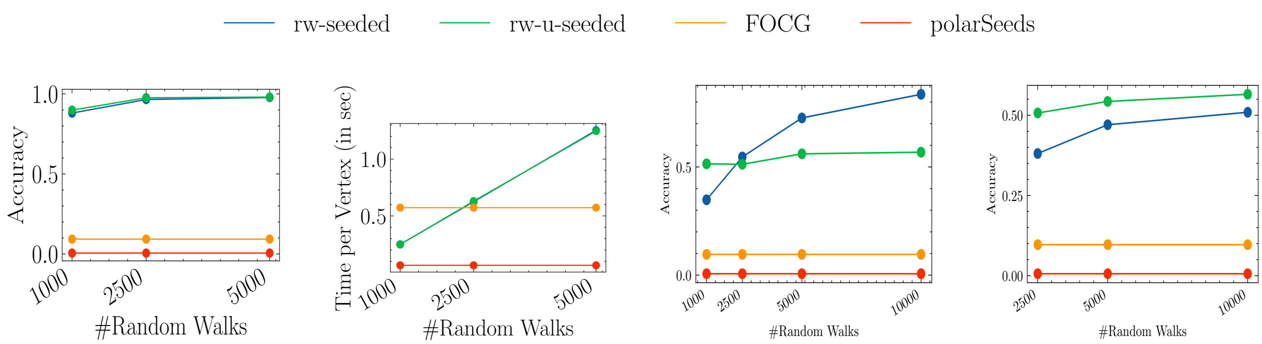
Table 1: Statistics for real-world datasets. Here, $|E^-|$ denotes the number of negative edges, $\overline{\deg}$ and $\deg_{\max}$ denote the average degree and maximum degree, and $|V|_{\text{labeled}}$ denotes the number of vertices with ground-truth labels.

Dataset	$\lvert V\rvert$	$\lvert E\rvert$	$\lvert E^-\rvert / \lvert E\rvert$	$\overline{\deg}$	$\deg_{\max}$	$\lvert V\rvert_{\text{labeled}}$
WikiS	9 211	395 038	0.39	140.3	1 503	9 211
WikiM	34 404	904 768	0.28	52.6	3 407	9 453
WikiL	258 259	3 187 096	0.08	24.7	6 017	9 540



(a) Vary #steps: clustering (b) Vary #walks: clustering (c) Vary #steps: biclustering (d) Vary #walks: biclustering

Figure 1: Accuracies of the algorithms on WikiS. We consider clustering (Figs. (a) and (b)) and biclustering (Figs. (c) and (d)) experiments, with varying number of steps (Figs. (a) and (c)) and varying number of random walks (Figs. (b) and (d)). Since FOCG and POLARSEEDS do not use the number of steps and random walks as input, we only ran them once.



(a) WikiL: clustering (b) WikiL: avg. query time (c) WikiM: biclustering (d) WikiL: biclustering

Figure 2: Results on WikiM and WikiL for varying numbers of random walks. We present accuracies for clustering on WikiL (Fig. (a)), and for biclustering in WikiM (Fig. (c)) and in WikiL (Fig. (d)). The running time per vertex for clustering in WikiL is given in Fig. (b). Since FOCG and POLARSEEDS do not use random walks as input, we only ran them once.

tions for this might be as follows: (1) For a node u, first identify the cluster U_i of u using the unsigned oracle and then use auxiliary information to decide whether u belongs to V_{2i-1} or to V_{2i}. (2) When the underlying graph contains few negative edges, bias the random walks of RW-SEEDED such that it takes disproportionately many negative edges.

Acknowledgements

This research is supported by the the ERC Advanced Grant REBOUND (834862), the EC H2020 RIA project SoBig-Data++ (871042), and the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation. P.P. is supported by "the Fundamental Research Funds for the Central Universities".

References

Atay, F. M. and Liu, S. Cheeger constants, structural balance, and spectral clustering analysis for signed graphs. *Discrete Mathematics*, 343(1):111616, 2020.

Bansal, N., Blum, A., and Chawla, S. Correlation clustering. *Mach. Learn.*, 56(1-3):89–113, 2004.

Bonchi, F., Galimberti, E., Gionis, A., Ordozgoiti, B., and Ruffo, G. Discovering polarized communities in signed networks. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pp. 961–970, 2019.

Cesa-Bianchi, N., Gentile, C., Vitale, F., and Zappella, G. A correlation clustering approach to link classification in signed networks. In Mannor, S., Srebro, N., and Williamson, R. C. (eds.), *COLT*, volume 23, pp. 34.1–34.20, 2012.

Chiang, K., Whang, J. J., and Dhillon, I. S. Scalable clustering of signed networks using balance normalized cut. In *CIKM*, pp. 615–624. ACM, 2012.

Chiang, K., Hsieh, C., Natarajan, N., Dhillon, I. S., and Tewari, A. Prediction and clustering in signed networks: a local to global perspective. *J. Mach. Learn. Res.*, 15(1): 1177–1213, 2014.

Chiplunkar, A., Kapralov, M., Khanna, S., Mousavifar, A., and Peres, Y. Testing graph clusterability: Algorithms and lower bounds. *CoRR*, abs/1808.04807, 2018. URL `http://arxiv.org/abs/1808.04807`. Conference version appeared in FOCS 2018.

Chu, L., Wang, Z., Pei, J., Wang, J., Zhao, Z., and Chen, E. Finding gangs in war from signed networks. In *KDD*, pp. 1505–1514, 2016.

Chung, F. R. *Spectral Graph Theory*. Number 92. American Mathematical Soc., 1997.

Csardi, G. and Nepusz, T. The igraph software package for complex network research. *InterJournal*, Complex Systems:1695, 2006. URL `https://igraph.org`.

Cucuringu, M., Davies, P., Glielmo, A., and Tyagi, H. SPONGE: A generalized eigenproblem for clustering signed networks. In *AISTATS*, volume 89, pp. 1088–1098. PMLR, 2019.

Cucuringu, M., Singh, A. V., Sulem, D., and Tyagi, H. Regularized spectral methods for clustering signed networks. *CoRR*, abs/2011.01737, 2020.

Czumaj, A., Peng, P., and Sohler, C. Testing cluster structure of graphs. *CoRR*, abs/1504.03294, 2015. URL `http://arxiv.org/abs/1504.03294`. Conference version appeared in STOC 2015.

Gharan, S. O. and Trevisan, L. Partitioning into expanders. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pp. 1256–1266. SIAM, 2014.

Gluch, G., Kapralov, M., Lattanzi, S., Mousavifar, A., and Sohler, C. Spectral clustering oracles in sublinear time. *CoRR*, abs/2101.05549, 2021. Conference version appeared in SODA 2021.

Harary, F. On the notion of balance of a signed graph. *Michigan Mathematical Journal*, 2(2):143–146, 1953.

Jung, J., Jin, W., Sael, L., and Kang, U. Personalized ranking in signed networks using signed random walk with restart. In *ICDM*, pp. 973–978, 2016.

Jung, J., Jin, W., and Kang, U. Random walk-based ranking in signed social networks: model and algorithms. *Knowl. Inf. Syst.*, 62(2):571–610, 2020.

Kunegis, J., Schmidt, S., Lommatzsch, A., Lerner, J., De Luca, E. W., and Albayrak, S. Spectral analysis of signed graphs for clustering, prediction and visualization. In *SDM*, pp. 559–570, 2010.

Mercado, P., Tudisco, F., and Hein, M. Clustering signed networks with the geometric mean of laplacians. In *NIPS*, pp. 4421–4429, 2016.

Mercado, P., Tudisco, F., and Hein, M. Spectral clustering of signed graphs via matrix power means. In *ICML*, pp. 4526–4536, 2019.

Ordozgoiti, B., Matakos, A., and Gionis, A. Finding large balanced subgraphs in signed networks. In *WWW*, pp. 1378–1388, 2020.

Peng, P. Robust Clustering Oracle and Local Reconstructor of Cluster Structure of Graphs. In *SODA*, pp. 2953–2972, 2020.

Pons, P. and Latapy, M. Computing communities in large networks using random walks. *J. Graph Algorithms Appl.*, 10(2):191–218, 2006.

Trevisan, L. Max cut and the smallest eigenvalue. *SIAM Journal on Computing*, 41(6):1769–1786, 2012.

Xiao, H., Ordozgoiti, B., and Gionis, A. Searching for polarization in signed graphs: a local spectral approach. In *WWW*, pp. 362–372, 2020.

A. Overview of the Appendix

The appendix is organized as follows:

- Appendix B: We provide further motivation for our choice of the inner signed bipartiteness ratio.
- Appendix C: We present the pseudocode for our algorithms.
- Appendix D: We give an overview of our proof strategy.
- Appendix E: We present the full proofs for all claims in the main text.
- Appendix F: We give details on the implementations of our algorithms, including parameter tuning.
- Appendix G: We evaluate our algorithm on synthetically generated data.

B. Further Motivation of the Inner Signed Bipartiteness Ratio

We provide further motivation for the inner signed bipartiteness ratio. Recall that we set

$$\beta^{\text{inner}}(G) := \min_{\emptyset \neq U \subseteq V:\ \text{vol}(U) \leq \frac{1}{2}\text{vol}(G)} \beta_G(U).$$

First, let us justify our intuition that if $\beta^{\text{inner}}(G)$ is large, then $\beta^{\text{inner}}(G)$ cannot be decomposed into two nearly-balanced communities. To make this more formal, recall the definition of $\beta_2(G) = \min_{U_1,U_2} \max_{i=1,2} \beta_G(U_i)$, where the minimum is taken over all partitions U_1, U_2 of V with $U_1, U_2 \neq \emptyset$. Now note that if we could split G into two nearly-balanced communities, then we would have that $\beta_2(G)$ is small. Thus, we show that if $\beta^{\text{inner}}(G)$ is large then $\beta_2(G)$ is at least as large. This justifies our informal intuition above.

Lemma 7. *It holds that $\beta_2(G) \geq \beta^{inner}(G)$. In particular, if $\beta_{in} \in \mathbb{R}_{\geq 0}$ and $\beta^{inner}(G) \geq \beta_{in}$ then $\beta_2(G) \geq \beta_{in}$.*

Proof. We prove the first claim by contradiction, i.e., suppose that $\beta^{\text{inner}}(G) = \beta_{\text{in}}$ and $\beta_2(G) < \beta_{\text{in}} = \beta^{\text{inner}}(G)$. Then there exists a partition U_1, U_2 of V such that $\beta_G(U_1) < \beta_{\text{in}}$ and $\beta_G(U_2) < \beta_{\text{in}}$. Next, since U_1 and U_2 form a partition of V, we must have that $\text{vol}(U_1) \leq \frac{1}{2}\text{vol}(G)$ or $\text{vol}(U_2) \leq \frac{1}{2}\text{vol}(G)$. This implies that there exists a set $U \in \{U_1, U_2\}$ such that $\text{vol}(U) \leq \frac{1}{2}\text{vol}(G)$ and $\beta_G(U) < \beta_{\text{in}}$. Thus, $\beta^{\text{inner}}(G) < \beta_{\text{in}}$. But this contradicts our assumption that $\beta^{\text{inner}}(G) = \beta_{\text{in}}$.

The second claim of the lemma immediately follows from applying the first claim. □

Why do we not assume that $\beta(G[U_i])$ is large? Next, we discuss why we cannot replace the inner signed bipartiteness ratio $\beta^{\text{inner}}(G)$ with the (classic) signed bipartiteness ratio $\beta(G[U_i])$ in Def. 3. To see this, consider Def. 3 with $\beta^{\text{inner}}(G[U_i]) \geq \beta_{\text{in}}$ replaced by $\beta(G[U_i]) \geq \beta_{\text{in}}$ for all $i = 1, \ldots, k$. Again, we consider the the setting with $\beta_{\text{in}} > \beta_{\text{out}}$. Intuitively, this would mean that subgraph $G[U_i]$ is "far from balanced" on its inside (since $\beta(G[U_i])$ is large) while outside it is close to balanced (since $\beta_G(U_i)$ is small).

Unfortunately, we show that if $\beta_{\text{in}} > \beta_{\text{out}}$ then no graph can satisfy this new definition. Before we give a more general proof, consider the following example. Consider a graph G which consists of two positive cliques among vertices V_1 and V_2 and in between V_1 and V_2 there is biclique consisting only of negative edges. Then $\beta(G) = 0$ and $\beta(V_1, V_2) = 0$ but also $\beta(G[V_1]) = 0$ and $\beta(G[V_2]) = 0$ (since graphs with only positive edges are balanced).

Now we give a more general result showing that for any $U \subseteq V$, we have that $\beta(G[U]) \leq \beta_G(U)$. Therefore, the previously proposed definition that avoids the inner signed bipartiteness ratio cannot work: it implies that $\beta_{\text{in}} \leq \beta(G[U_i]) \leq \beta_G(U_i) \leq \beta_{\text{out}}$ but this contradicts our assumption that $\beta_{\text{in}} > \beta_{\text{out}}$.

Lemma 8. *For any two disjoint subsets $V_1, V_2 \subseteq V$, it holds that $\beta(G[V_1 \cup V_2]) \leq \beta_G(V_1, V_2)$. Furthermore, for any $U \subseteq V$, it holds that $\beta(G[U]) \leq \beta_G(U)$.*

Proof. We have that

$$\begin{aligned}
&\beta(G[V_1 \cup V_2]) \\
&\leq \beta_{G[V_1 \cup V_2]}(V_1, V_2) \\
&= \frac{2\left|E_G^+(V_1, V_2)\right| + \left|E_G^-(V_1)\right| + \left|E_G^-(V_2)\right|}{\text{vol}_{G[V_1 \cup V_2]}(V_1, V_2)}
\end{aligned}$$

$$
\begin{aligned}
&\leq \frac{2\left|E_G^+(V_1,V_2)\right| + \left|E_G^-(V_1)\right| + \left|E_G^-(V_2)\right| + \left|E_G(V_1\cup V_2, \overline{V_1 \cup V_2})\right|}{\mathrm{vol}_{G[V_1\cup V_2]}(V_1,V_2) + \left|E_G(V_1\cup V_2, \overline{V_1\cup V_2})\right|} \\
&= \beta_G(V_1,V_2),
\end{aligned}
$$

where we have used that for $a, b, c > 0$ and $a \leq b$, it holds that $\frac{a}{b} \leq \frac{a+c}{b+c}$.

Now for any subset U, let V_1, V_2 be a partition of U such that $\beta_G(U) = \beta_G(V_1, V_2)$. Then by the above calculation, we have that

$$\beta(G[U]) \leq \beta_{G[U]}(V_1,V_2) \leq \beta_G(V_1,V_2) = \beta_G(U). \qquad \square$$

Is the underlying unsigned graph clusterable? Next, we argue that our condition from Def. 3 is not implied by previous definitions that were based on the conductance of *unsigned* graphs (e.g., in (Czumaj et al., 2015)). In other words, it is possible that our methods finds the planted clusters in the signed graph while this would not be possible by only looking at the underlying unsigned graph.

First, recall that for an unsigned graph $G^{\text{un}} = (V, E)$ and a set of vertices $S \subseteq V$, the *conductance* of S is given by $\phi_{G^{\text{un}}}(S) = \frac{|E(S, V\setminus S)|}{\mathrm{vol}(S)}$. Furthermore, we let $\phi(G^{\text{un}})$ denote the conductance of G^{un} which is given by $\phi(G^{\text{un}}) = \min\{\phi_{G^{\text{un}}}(S)\colon\ \mathrm{vol}(S) \leq \mathrm{vol}(G^{\text{un}})/2\}$.

Now consider the unsigned graph G^{un} that is obtained by removing all the signs on the edges from G. We argue that a $(k, \beta_{\text{in}}, \beta_{\text{out}})$-clustering of G does not imply a $(k, \beta_{\text{in}}, \beta_{\text{out}})$-clustering of G^{un}, i.e., G^{un} contains a k-partition $C_1, \ldots, C_k$ such that the *inner conductance* of C_i, denoted $\phi(G^{\text{un}}[C_i])$, is at least β_{in} and the *outer conductance* of C_i, denoted $\phi_G(C_i)$, is at most β_{out} for each i. Therefore, *one could* ***not*** *apply the previous clustering oracle for conductance-based clustering of* G^{un} *and recover the underlying clusters in our problem.* We give a brief explanation next.

Suppose that $U_1, \ldots, U_k$ is a $(k, \beta_{\text{in}}, \beta_{\text{out}})$-clustering of G. Then, indeed, it is true that each U_i has small outer conductance in G^{un}, since we have that $\phi_{G^{\text{un}}}(U_i) \leq \beta_G(U_i) \leq \beta_{\text{out}}$. However, the inner conductance of U_i in G^{un} can be arbitrarily small: Even though the inner signed bipartiteness ratio of U_i is large, it can happen that there is a very small subset $S_i \subseteq U_i$ (say, of size $O(\log n)$) such that there are almost no edges leaving S_i but all edges in S_i have sign $-$ and S_i is far from being balanced. Thus, there exists a subset S_i in U_i whose (outer) conductance is almost 0 in G^{un} but the inner signed bipartiteness ratio of S_i and U_i is large.

Note that the previous example with the set S_i also shows that it is possible that G contains a sparse cut and, therefore, $\mathbf{p}_u^t$ does not converge to the uniform distribution of V.

C. Pseudocode for Our Algorithms

Algorithm 1 Estimating the dot product $\langle \mathbf{p}_u^t \mathbf{D}^{-1/2}, \mathbf{p}_v^t \mathbf{D}^{-1/2} \rangle$

procedure ESTDOTPROD(u, v, t, α)
  $R \leftarrow \frac{40000 d^2 k^{1.5} \sqrt{n}}{\alpha^{1.5}}$
  for all $i = 1, \cdots, h = O(\log n)$ **do**
    for $x \in \{u, v\}$ **do**
      Perform R lazy signed random walks of length t starting at vertex x with sign $+$.
      for each $w \in V$ **do**
        $\mathbf{m}_x^+(w) \leftarrow$ the fraction of walks that end at w with sign $+$.
        $\mathbf{m}_x^-(w) \leftarrow$ the fraction of walks that end at w with sign $-$.
        $\mathbf{m}_x \leftarrow (\mathbf{m}_x^+ - \mathbf{m}_x^-)\mathbf{D}^{-1/2}$.
    $\chi_i \leftarrow \langle \mathbf{m}_u, \mathbf{m}_v \rangle$.
  Let X_{uv} be the median value of $\chi_1, \cdots, \chi_h$.
  return X_{uv}

Algorithm 2 Preprocessing: Constructing a clustering oracle

procedure BUILDORACLE$(G, k, d, \beta_{\text{in}}, \epsilon, \gamma)$
    $s \leftarrow \frac{20k\log(k)}{\gamma}$, $\alpha \leftarrow \frac{\epsilon}{90s}$, $t \leftarrow \frac{C'' k^6 d^3 \log n}{\epsilon \cdot \beta_{\text{in}}^2}$ for constant C''
    Sample a set S of s vertices independently and uniformly at random from V.
    for $v \in S$ **do** ▷ test if $||\mathbf{p}_v^t \mathbf{D}^{-1/2}||_2^2 = O(\frac{k^2 \log(k)}{\gamma\epsilon \cdot n})$
        $X_{vv} \leftarrow$ESTDOTPROD(v, v, t, α)
        if $X_{vv} \geq \frac{4000k^2\log(k)}{\gamma\epsilon\cdot n}$ **then**
            abort and **return Fail**
    let H be an empty graph with vertex set S
    for each pair $u, v \in S$ **do** ▷ test if $\Delta_{uv} \leq \frac{1}{4nd}$
        $X_{vv} \leftarrow$ESTDOTPROD(v, v, t, α)
        $X_{uu} \leftarrow$ESTDOTPROD(u, u, t, α)
        $X_{uv} \leftarrow$ESTDOTPROD(u, v, t, α)
        $\delta_{uv} \leftarrow \min\{X_{vv} + X_{uu} - 2X_{uv}, X_{vv} + X_{uu} + 2X_{uv}\}$
        if $\delta_{uv} \leq \frac{1}{2dn}$ **then**
            add edge (u, v) to H
    if H is the union of k connected components (CCs) **then**
        label the components by "$1, 2, \ldots, k$'
        label each vertex $u \in V_H$ with the same index as its component, denoted $\ell(u)$
        return H and its vertex labeling ℓ
    else
        return Fail

Algorithm 3 Answering the community membership of a vertex v

procedure WHICHCLUSTER(G, v, H, ℓ)
    for $u \in V_H$ **do**
        $X_{vv} \leftarrow$ESTDOTPROD(v, v, t, α)
        $X_{uu} \leftarrow$ESTDOTPROD(u, u, t, α)
        $X_{uv} \leftarrow$ESTDOTPROD(u, v, t, α)
        $\delta_{uv} \leftarrow \min\{X_{vv} + X_{uu} - 2X_{uv}, X_{vv} + X_{uu} + 2X_{uv}\}$
        if $\delta_{uv} \leq \frac{1}{2dn}$ **then**
            abort and **return** the label $\ell(u)$
    return a random number from $\{1, 2, \ldots, k\}$

D. Analysis Overview

In this section, we give an overview of the analysis and provide the main technical lemmas of our analysis. All missing proofs can be found in App. E.

Intuition. We begin by providing some intuition for our algorithm and our analysis.

We start by establishing some properties of lazy signed random walks. First, suppose that we perform the lazy random walks *without the signs* on the underlying unsigned graph G^{un} with the corresponding transition probability matrix $\mathbf{W}^{\text{un}} := \frac{\mathbf{I}+\mathbf{D}^{-1}\mathbf{A}^{\text{un}}}{2}$, where $\mathbf{A}^{\text{un}}$ is the adjacency matrix of G^{un}. Then the probability that a random walk started at vertex u ends in vertex v is $\mathbf{Pr}\left(v_t = v\right) = \mathbf{q}_u^t(v)$, where $\mathbf{q}_u^t = \mathbf{1}_u(\mathbf{W}^{\text{un}})^t$. Second, when we *add the sign*, then we are interested in the quantity

$$\mathbf{p}_u^t(v) = \mathbf{Pr}\left(v_t = v, s_t = +\right) - \mathbf{Pr}\left(v_t = v, s_t = -\right),$$

i.e., $\mathbf{p}_u^t(v)$ is the probability of reaching v with a positive sign *minus* the probability of reaching v with a negative sign. Observe that $\mathbf{p}_u^t(v)$ can be described by the walk matrix $\mathbf{W} = \frac{\mathbf{I}+\mathbf{D}^{-1}\mathbf{A}^{\sigma}}{2}$, i.e., $\mathbf{p}_u^t(v) = [\mathbf{1}_u\mathbf{W}^t](v)$ for all v. Note that while $\mathbf{q}_u^t$ (for unsigned random walks) gives a distribution, this is not the case for $\mathbf{p}_u^t$. In fact, $\mathbf{p}_u^t$ can potentially even contain negative entries and $\mathbf{p}_u^t$ does not necessarily converge to the uniform distribution of V as it can happen that V contains a sparse cut (we discuss this in App. B). In the following, we call such a vector $\mathbf{p}_u^t$ the *discrepancy vector* of a lazy signed random walk of length t starting from u.

Next, consider a signed graph G and a $(k, \beta_{\text{in}}, \beta_{\text{out}})$-clustering $U_1, \dots, U_k$ of G as per Def. 3. Let $U \in \{U_1, \dots, U_k\}$ be one of the clusters. Since by assumption we have that $\beta_G(U) \le \beta_{\text{out}}$, there exists a partition of U into subsets V_1 and V_2 with $\beta_G(V_1, V_2) \le \beta_{\text{out}}$. Then, intuitively, for a typical vertex $u \in U$, a *short* random walk starting from u of has the following properties:

(i) Since the walk is short (this is crucial), the walk will be "trapped" in U, as there are only few edges leaving U. Thus, for the discrepancy vector it should hold that $\sum_{v \in U} |\mathbf{p}_u^t(v)| \gg \sum_{v \in V \setminus U} |\mathbf{p}_u^t(v)|$.

(ii) If $u \in V_1$ then most walks ending at vertices $v \in V_1$ should have a positive sign and most walks ending at vertices $V \in V_2$ should have negative sign. Thus, for the discrepancy vector $\mathbf{p}_u^t$ it should hold that $\mathbf{p}_u^t(v) > 0$ if $v \in V_1$ and $\mathbf{p}_u^t(v) < 0$ if $v \in V_2$. Similarly, if $u \in V_2$ then the same holds with flipped signs.

(iii) Let $u \in U$. If two vertices x, y are from the same sub-communities (i.e., $x, y \in V_1$ or $x, y \in V_2$), then $\mathbf{p}_u^t(x) \approx \mathbf{p}_u^t(y)$, i.e., the discrepancy on x is close to the discrepancy on y.

Now let us discuss how we can use the discrepancy vectors $\mathbf{p}_u^t$ and $\mathbf{p}_v^t$ to decide whether u and v are in the same cluster or not. Our goal is to find a distance function $\Delta_{uv} = \Delta_{uv}(\mathbf{p}_u^t, \mathbf{p}_v^t)$ such that Δ_{uv} is small iff u and v are from the same cluster U.

First, consider vertices $u \in U_i$ and $v \in U_j$, $i \neq j$, from different clusters. Then Property (i) suggests that $\mathbf{p}_u^t$ and $\mathbf{p}_v^t$ have most of their mass in different entries and thus $\|\mathbf{p}_u^t - \mathbf{p}_v^t\|_2^2$ should be large.

Second, consider vertices $u, v \in U$ from the same cluster. Let $U = V_1 \cup V_2$ as above. If $u, v \in V_1$ or $u, v \in V_2$ then the properties above suggest $\mathbf{p}_u^t \approx \mathbf{p}_v^t$ and thus $\|\mathbf{p}_u^t - \mathbf{p}_v^t\|_2^2 \approx 0$ is small. However, if $u \in V_1$ and $v \in V_2$, then $\mathbf{p}_u^t \approx -\mathbf{p}_v^t$ by Property (ii) and (iii) and thus $\|\mathbf{p}_u^t - \mathbf{p}_v^t\|_2^2 \approx \|2\mathbf{p}_u^t\|_2^2$ will still be large. However, this issue can be mitigated if we use $\mathbf{r}_u^t = |\mathbf{p}_u^t|$ and $\mathbf{r}_v^t = |\mathbf{p}_v^t|$ instead of $\mathbf{p}_u^t$ and $\mathbf{p}_v^t$. Here, the absolute values are applied component-wise, i.e., $\mathbf{r}_u^t(v) = |\mathbf{p}_u^t(v)|$ for all v.

Therefore, in our analysis we would like to use $\|\mathbf{r}_u^t - \mathbf{r}_v^t\|_2^2$ as a distance measure. However, due to some technical difficulties in the analysis of this quantity, we cannot use the vectors $\mathbf{r}_u^t$ and instead we consider $\min\{\|\mathbf{p}_u^t - \mathbf{p}_v^t\|_2^2, \|\mathbf{p}_u^t + \mathbf{p}_v^t\|_2^2\}$ which behaves similar to taking the absolute values and also fixes the issue with Property (ii). After adding some degree corrections, we arrive at our final distance measure $\Delta_{uv} = \min\{\|\mathbf{p}_u^t \mathbf{D}^{-1/2} - \mathbf{p}_v^t \mathbf{D}^{-1/2}\|_2^2, \|\mathbf{p}_u^t \mathbf{D}^{-1/2} + \mathbf{p}_v^t \mathbf{D}^{-1/2}\|_2^2\}$. Note that Δ is a pseudometric distance, i.e., one may have that $\Delta_{uv} = 0$ for distinct vectors $\mathbf{p}_u^t$ and $\mathbf{p}_v^t$.

D.1. Main Technical Lemmas

We give a technical overview of our analysis. Note that while our high-level proof strategy is relatively similar to the one used in (Czumaj et al., 2015) for unsigned graphs, the concrete proofs are often quite different. It required a substantial amount of work and new ideas to obtain our results for signed graphs.

Random walks from the same cluster. First, we show that if U is a polarized community with large inner signed bipartiteness ratio and small (outer) bipartiteness ratio, then for most of the vertices $u, v \in U$ their distance Δ_{uv} is small.

Lemma 9. *Let $\alpha, \gamma \in (0, 1)$. Let $G = (V, E, \sigma)$ be a signed, d-bounded degree, $(k, \beta_{in}, \beta_{out})$-clusterable graph. Let U be a subset of V such that $|U| \ge \gamma n$, $\beta^{inner}(G[U]) \ge \beta_{in}$ and $\beta_G(U) \le \beta_{out}$. Then for all $t \ge \frac{4C_1^2(k+1)^6 \log n}{\beta_{in}^2}$, $\beta_{out} < \alpha_1 \beta_{in}^2$ where $\alpha_1 = \frac{\alpha\gamma}{800000 C_1^2 k d^3}$, there exists a subset $\widetilde{U} \subseteq U$ with $\left|\widetilde{U}\right| \ge (1-\alpha)\,|U|$ and for all $u, v \in \widetilde{U}$, it holds that $\Delta_{uv} \le \frac{1}{4nd}$.*

Proving Lem. 9 was one of the main obstacles for obtaining our results. We will give an overview of its proof and our main technical contribution in Sec. 3.2.

Random walks from two different clusters. We further show that if we have two disjoint communities U_1 and U_2, then for most vertices $u \in U_1$ and $v \in U_2$ their distance Δ_{uv} must be large. Note that while Lem. 9 assumes a lower bound on the walk length t, Lem. 10 assumes an upper bound on t. Thus, it is crucial that the random walks have the correct length $\Theta(\log n)$.

Lemma 10. *Let U_1 and U_2 be two disjoint subsets with $\beta_G(U_1), \beta_G(U_2) \le \beta_{out}$. Let $0 < \alpha < 1$. For any $0 \le t \le \frac{\alpha}{8\beta_{out}}$, there exist subsets $\widehat{U_1} \subseteq U_1, \widehat{U_2} \subseteq U_2$ such that $|\widehat{U_1}| \ge (1-\alpha)|U_1|$, $|\widehat{U_2}| \ge (1-\alpha)|U_2|$, and for any $u \in \widehat{U_1}$ and $v \in \widehat{U_2}$,* *it holds that* $\Delta_{uv} \geq \frac{1}{nd}$.

ℓ_2^2-norm of the vector $\mathbf{p}_v^t\mathbf{D}^{-1/2}$. Based on the previous two lemmas, our goal will be to test whether $\Delta_{uv} \leq \frac{1}{4nd}$ or $\Delta_{uv} \geq \frac{1}{nd}$. To this end, we wish to use ESTDOTPROD to approximate Δ_{uv} via Eqn.s (2) and (3). To prove this, we first give a useful bound on the ℓ_2^2-norm of the vectors $\mathbf{p}_v^t\mathbf{D}^{-1/2}$.

Lemma 11. *Let $\alpha \in (0,1)$. Suppose $G = (V, E, \sigma)$ is a signed and $(k, \beta_{in}, \beta_{out})$-clusterable graph. Then there exists a set $V' \subseteq V$ of size $|V'| \geq (1-\alpha)|V|$ such that for all $u \in V'$ and all $t \geq \frac{C_1^2(k+1)^6 \log n}{\beta_{in}^2}$, we have that $\|\mathbf{p}_v^t\mathbf{D}^{-1/2}\|_2^2 \leq \frac{2k}{\alpha n}$.*

Estimating the dot product. Now we prove that ESTDOTPROD(u,v,t,α) estimates $\langle \mathbf{p}_u^t\mathbf{D}^{-1/2}, \mathbf{p}_v^t\mathbf{D}^{-1/2}\rangle$ with small error.

Lemma 12. *Let $\alpha \in (0,1)$ be a number such that $\frac{2k}{\alpha} \leq n$. Suppose $G = (V, E, \sigma)$ is a signed and $(k, \beta_{in}, \beta_{out})$-clusterable graph. Let $t \geq \frac{C_1^2(k+1)^6 \log n}{\beta_{in}^2}$. Let $V' \subseteq V$ be the set of vertices satisfying the property given by Lem. 11. Then* ESTDOTPROD*(u, v, t, α) outputs X_{uv} such that with probability $1 - 1/n^3$, it holds that*

$$\left| X_{uv} - \langle \mathbf{p}_v^t\mathbf{D}^{-1/2}, \mathbf{p}_u^t\mathbf{D}^{-1/2}\rangle \right| \leq \frac{1}{20nd}$$

for all $u, v \in V'$. Furthermore, ESTDOTPROD*(u, v, t, α) runs in time $O(\frac{d^2k^{1.5}t\log n}{\alpha^{1.5}} \cdot \sqrt{n})$.*

To prove Thm. 4, we first show that an overwhelming fraction of the vertices are "well-behaved" in the senses of Lem.s 9–11. Then, if we only consider these "well-behaved" vertices, we can apply Lem. 12 and this will classify all of these vertices correctly with high probability.

E. Deferred Proofs

E.1. Proof of Lemma 5

Now we prove Lem. 5, which is restated in the following for the sake of readability.

Lemma 5. *If G is signed and $(k, \beta_{in}, \beta_{out})$-clusterable, then $\lambda_i \leq 2\beta_{out}$ for all $i \leq k$ and $\lambda_i \geq \frac{\beta_{in}^2}{C_1^2(k+1)^6}$ for all $i \geq k+1$, where C_1 is the constant from Thm. 2.*

Proof of Lemma 5. Since G is $(k, \beta_{\text{in}}, \beta_{\text{out}})$-clusterable, there exists a partition of V into clusters $(C_1, \ldots, C_k)$ such that $\beta^{\text{inner}}(G[C_i]) \geq \beta_{\text{in}}$ and $\beta_G(C_i) \leq \beta_{\text{out}}$ for all $i \in [k]$. Therefore, $\beta_k(G) \leq \max_i \beta_G(C_i) \leq \beta_{\text{out}}$. Now Thm. 2 implies that $\lambda_k \leq 2\beta_k(G) \leq 2\beta_{\text{out}}$. Thus, $\lambda_1 \leq \cdots \leq \lambda_k \leq 2\beta_{\text{out}}$.

The next part shows that $\lambda_{k+1} \geq \frac{\beta_{\text{in}}^2}{C_1^2(k+1)^6}$. Consider any $k+1$ disjoint subsets $V_1, \ldots, V_{k+1}$.

Now observe that there must exist a subset $V_{i_0} \in \{V_1, \ldots, V_{k+1}\}$ with following property: for all $i \in [k]$, $\text{vol}(V_{i_0} \cap C_i) \leq \frac{1}{2}\text{vol}(C_i)$. To see that this is the case, suppose that the statement is false, i.e., no such subset exists. Then by pigeonhole principle there must exist indices j_1, j_2 with $j_1 \neq j_2$ and index j_3 such that $V_{j_1} \cap C_{j_3}$ and $V_{j_2} \cap C_{j_3}$ have volume more than $\frac{1}{2}\text{vol}(C_{j_3})$. This is a contradiction since the sets V_{j_1} and V_{j_2} are mutually disjoint.

For the rest of the proof, consider the subset V_{i_0} with the above property. Consider an arbitrary partition L, R of V_{i_0}, such that $L \cup R = V_{i_0}$ and $L \cap R = \emptyset$. Let $L_i = C_i \cap L$ and $R_i = C_i \cap R$. Observe that $V_{i_0} = \bigcup_i (L_i \cup R_i)$ since $V = \bigcup_i C_i$. Since $\beta^{\text{inner}}(G[C_i]) \geq \beta_{\text{in}}$ and $\text{vol}(L_i \cup R_i) \leq \frac{1}{2}\text{vol}(C_i)$ for all $i \leq k$, we get that $\beta_{G[C_i]}(L_i \cup R_i) \geq \beta_{\text{in}}$ for all $i \in [k]$. In particular, this implies that $\beta_{G[C_i]}(L_i, R_i) = \frac{e_{G[C_i]}(L_i, R_i)}{\text{vol}(L_i \cup R_i)} \geq \beta_{\text{in}}$ for all $i \in [k]$. Thus,

$$\begin{aligned}
\beta_G(L, R) &= \frac{e_G(L, R)}{\text{vol}(L \cup R)} \\
&= \frac{2\left|E_G^+(L, R)\right| + \left|E_G^-(L)\right| + \left|E_G^-(R)\right| + \left|E_G(L \cup R, V \setminus (L \cup R))\right|}{\text{vol}(L \cup R)} \\
&\geq \frac{1}{\text{vol}(L \cup R)} \sum_{i \in [k]} \left[2\left|E_G^+(L_i, R_i)\right| + \left|E_G^-(L_i)\right| + \left|E_G^-(R_i)\right| + \left|E_G(L_i \cup R_i, C_i \setminus (L_i \cup R_i))\right| \right]
\end{aligned}$$

$$= \frac{\sum_{i\in[k]} e_{G[C_i]}(L_i, R_i)}{\sum_{i\in[k]} \mathrm{vol}(L_i \cup R_i)}$$
$$\geq \frac{\sum_{i\in[k]} \beta_{\text{in}} \mathrm{vol}(L_i \cup R_i)}{\sum_{i\in[k]} \mathrm{vol}(L_i \cup R_i)}$$
$$\geq \beta_{\text{in}}.$$

Since L, R is an arbitrary partition of V_{i_0}, it holds that $\beta_G(V_{i_0}) \geq \beta_{\text{in}}$. Thus, $\beta_{k+1}(G) \geq \beta_{\text{in}}$. Now Thm. 2 implies that $\lambda_{k+1} \geq \frac{\beta_{\text{in}}^2}{C_1^2(k+1)^6}$. □

E.2. Proof of Lemma 6

Now we prove Lem. 6, which is restated in the following for the sake of readability.

Lemma 6. *Let $\alpha \in (0, 1)$. Suppose $G = (V, E, \sigma)$ is signed and $(k, \beta_{in}, \beta_{out})$-clusterable. Let U be a cluster of G with $\beta^{inner}(G[U]) \geq \beta_{in}$ and $\beta_G(U) \leq \beta_{out}$. Then there exists a partition V_1, V_2 of U, a subset $\widetilde{U}$ of U, and constants $c_i, 1 \leq i \leq k$, such that $|c_i| \leq 3d$, $|\widetilde{U}| \geq (1-\alpha)|U|$, and for each $i \leq k$,*

- *if $u \in V_1 \cap \widetilde{U}$, then $\left|\mathbf{v}_i'(u) - c_i \cdot \frac{1}{\sqrt{\mu_U}}\right| \leq \frac{64dC_1}{\beta_{in}} \cdot \sqrt{\frac{\beta_{out}}{\alpha \cdot \mu_U}}$*
- *if $u \in V_2 \cap \widetilde{U}$, then $\left|\mathbf{v}_i'(u) + c_i \cdot \frac{1}{\sqrt{\mu_U}}\right| \leq \frac{64dC_1}{\beta_{in}} \cdot \sqrt{\frac{\beta_{out}}{\alpha \cdot \mu_U}}$*

where C_1 is the constant from Thm. 2.

Proof. The proof is based on the following intuition. Recall the definitions of vectors $\mathbf{v}_i', \mathbf{1}_{V_1,V_2}, \mathbf{w}_1$ from the above discussion. Since both $\mathbf{v}_i'$ and a scalar multiplications of $\mathbf{1}_{V_1,V_2}$ have small total 'discrepancy' over the set of all edges (i.e., Ineq. (4) and (7)), and the ratio between the total 'discrepancy' over all the edges and the total 'discrepancy' over all vertices (w.r.t. some centers defined by $\mathbf{w}_1$) is large (i.e., Ineq. (8)), one can guarantee both that $\mathbf{v}_i'$ and a scaled multiplication of $\mathbf{1}_{V_1,V_2}$ are close to (another scalar multiplication of) $\mathbf{w}_1$. We now give the details.

Since G is $(k, \beta_{\text{in}}, \beta_{\text{out}})$-clusterable, we can apply Lem. 5 to obtain $\lambda_{k+1} \geq \frac{\beta_{\text{in}}^2}{C_1^2(k+1)^6}$ and $\lambda_i \leq 2\beta_{\text{out}}$ for all $i \in [k]$.

Since for any $i \leq k$, $\mathbf{v}_i \mathcal{L}^\sigma = \lambda_i \mathbf{v}_i$, we have $\mathbf{v}_i \mathbf{D}^{-1/2}(\mathbf{D} - \mathbf{A}^\sigma)\mathbf{D}^{-1/2} = \lambda_i \mathbf{v}_i$, and thus

$$\begin{aligned}\lambda_i &= \mathbf{v}_i \mathbf{D}^{-1/2}(\mathbf{D} - \mathbf{A}^\sigma)\mathbf{D}^{-1/2}\mathbf{v}_i^\top \\ &= \mathbf{v}_i'(\mathbf{D} - \mathbf{A}^\sigma)\mathbf{v}_i'^\top \\ &= \sum_{(u,v)\in E} \left(\mathbf{v}_i'(u) - \sigma(uv)\mathbf{v}_i'(v)\right)^2.\end{aligned}$$

Thus, for any $i \leq k$,

$$\sum_{(u,v)\in E} \left(\mathbf{v}_i'(u) - \sigma(uv)\mathbf{v}_i'(v)\right)^2 \leq 2\beta_{\text{out}}. \tag{4}$$

Now consider the subgraph $H = G[U]$ induced by the subset U. Denote the set of vertices of H as $V_H = U$ and the set of edges of H as E_H. By assumption on U, $\beta^{\text{inner}}(H) \geq \beta_{\text{in}}$ and $\beta_G(V[H]) \leq \beta_{\text{out}}$.

Let $\lambda_1(H), \lambda_2(H)$ be the first and second eigenvalues of the normalized Laplacian matrix $\mathcal{L}_H^\sigma$ of H. Next, let $\mathbf{D}_H$ be the degree matrix of H and let $\mathbf{A}_H^\sigma$ denote the signed adjacency matrix of H.

Now we consider a partition (V_1, V_2) of U such that $\beta_G(U) = \beta_G(V_1, V_2) \leq \beta_{\text{out}}$. Then it holds that $\beta(H) \leq \beta_H(V_1, V_2) \leq \beta_G(V_1, V_2) \leq \beta_{\text{out}}$. Therefore, by Thm. 2,

$$\lambda_1(H) \leq 2\beta_{\text{out}}.$$

Let $\mathbf{1}_{V_1,V_2} \in \mathbb{R}^{V_H}$ such that $\mathbf{1}_{V_1,V_2}(u) = 1$ if $u \in V_1$ and -1 otherwise. Set $\mathbf{b} := \frac{\mathbf{1}_{V_1,V_2}}{\sqrt{\mu_U}}$. Then it holds that

$$\sum_{(u,v)\in E} (\mathbf{b}(u) - \sigma(uv)\mathbf{b}(v))^2 = \frac{\mathbf{1}_{V_1,V_2}}{\sqrt{\mu_U}}(\mathbf{D}_H - \mathbf{A}_H^\sigma)\frac{\mathbf{1}_{V_1,V_2}{}^\top}{\sqrt{\mu_U}}$$

$$
\begin{aligned}
&= \frac{1}{\mu_U} \sum_{(u,v)\in E(H)} (\mathbf{1}_{V_1,V_2}(u) - \sigma(u,v)\mathbf{1}_{V_1,V_2}(v))^2 \\
&= \frac{2|E^+(V_1,V_2)| + |E^-(V_1)| + |E^-(V_2)|}{\mu_U} && (5)\\
&= \beta_H(V_1,V_2) && (6)\\
&\le \beta_{\text{out}}. && (7)
\end{aligned}
$$

Let $\mathbf{w}_1 \in \mathbb{R}^{V_H}$ be an eigenvector corresponding to $\lambda_1(H)$ such that $\mathbf{w}_1\mathbf{w}_1^\top = 1$, then it holds that $\mathbf{w}_1\mathcal{L}_H^\sigma = \lambda_1(H)\mathbf{w}_1$. Now since $\beta^{\text{inner}}(H) = \beta^{\text{inner}}(G[U]) \ge \beta_{\text{in}}$, we know (see App. B) that $\beta_2(H) = \beta_2(G[U]) \ge \beta^{\text{inner}}(G[U]) \ge \beta_{\text{in}}$. Thus, by Thm. 2,

$$
\lambda_2(H) \ge \frac{\beta_2(H)^2}{64C_1^2} \ge \frac{\beta_{\text{in}}^2}{64C_1^2}.
$$

Now by the variational characterization of the eigenvalues (see, e.g., Eqn. (1.7) in (Chung, 1997)), we have

$$
\lambda_2(H) = \min_{\mathbf{f}\in\mathbb{R}^{V_H}} \max_{c\in\mathbb{R}} \frac{\sum_{(u,v)\in E_H}(\mathbf{f}(u) - \sigma(uv)\mathbf{f}(v))^2}{\sum_u(\mathbf{f}(u) - c\cdot\mathbf{w}_1(u))^2 d_H(u)}. \tag{8}
$$

Now for any $i \le k$, if we let $\mathbf{f} = \mathbf{v}'_{i|H}$, i.e., the function $\mathbf{v}'_i$ that is restricted on H, in Inequality (8), and let

$$
c_{1,i} := \arg\min_{c\in\mathbb{R}:c\neq 0} \sum_u (\mathbf{f}(u) - c\cdot\mathbf{w}_1(u))^2 d_H(u),
$$

then we have that

$$
\frac{\sum_{(u,v)\in E_H}(\mathbf{v}'_i(u) - \sigma(uv)\mathbf{v}'_i(v))^2}{\sum_u(\mathbf{v}'_i(u) - c_{1,i}\cdot\mathbf{w}_1(u))^2 d_H(u)} \ge \lambda_2(H) \ge \frac{\beta_{\text{in}}^2}{64C_1^2}. \tag{9}
$$

If we let $\mathbf{f} = \mathbf{b}$ in Inequality (8), and let

$$
c_2 := \arg\min_{c\in\mathbb{R}:c\neq 0} \sum_u (\mathbf{b}(u) - c\cdot\mathbf{w}_1(u))^2 d_H(u).
$$

Thus,

$$
\frac{\sum_{(u,v)\in E_H}(\mathbf{b}(u) - \sigma(uv)\mathbf{b}(v))^2}{\sum_u(\mathbf{b}(u) - c_2\cdot\mathbf{w}_1(u))^2 d_H(u)} \ge \lambda_2(H) \ge \frac{\beta_{\text{in}}^2}{64C_1^2}. \tag{10}
$$

Therefore, by Inequalities (4), (7), (9) and (10), it holds that

$$
\begin{aligned}
&\sum_u (\mathbf{v}'_i(u) - c_{1,i}\cdot\mathbf{w}_1(u))^2 d_H(u) \le \frac{128C_1^2\beta_{\text{out}}}{\beta_{\text{in}}^2}, \text{ and} \\
&\sum_u (\mathbf{b}(u) - c_2\cdot\mathbf{w}_1(u))^2 d_H(u) < \frac{128C_1^2\beta_{\text{out}}}{\beta_{\text{in}}^2}.
\end{aligned} \tag{11}
$$

Equivalently,

$$
\begin{aligned}
&\|\mathbf{v}'_i\mathbf{D}^{1/2} - c_{1,i}\mathbf{w}_1\mathbf{D}^{1/2}\|_2^2 \le \frac{128C_1^2\beta_{\text{out}}}{\beta_{\text{in}}^2} < \frac{1}{4}, \text{ and} \\
&\|c_2\mathbf{w}_1\mathbf{D}^{1/2} - \mathbf{b}\mathbf{D}^{1/2}\|_2^2 \le \frac{128C_1^2\beta_{\text{out}}}{\beta_{\text{in}}^2} < \frac{1}{4},
\end{aligned}
$$

where we make use the fact that $512C_1^2\beta_{\text{out}} < \beta_{\text{in}}^2$.

Recall that $\mathbf{v}'_i = \mathbf{v}_i\mathbf{D}^{-1/2}$ and $\mathbf{b} = \frac{\mathbf{1}_{V_1,V_2}}{\sqrt{\mu_U}}$. Therefore,

$$\begin{aligned}\|\mathbf{v}'_i\mathbf{D}^{1/2}\|_2^2 &= \sum_{u\in V_H} {\mathbf{v}'_i}^2(u)d_H(u)\\ &= \sum_{u\in V_H} {\mathbf{v}_i}^2(u)d_H(u)^{-1}\cdot d_H(u)\\ &\le \sum_{u\in V}\mathbf{v}_i^2(u)\\ &= 1\end{aligned}$$

and

$$\|\mathbf{b}\mathbf{D}^{1/2}\|_2^2 = \sum_{u\in V_H}\mathbf{b}^2(u)d_H(u) = 1.$$

By the above inequalities, we have

$$|c_{1,i}|\cdot\|\mathbf{w}_1\mathbf{D}^{1/2}\|_2 \le \frac{1}{2} + \|\mathbf{v}'_i\mathbf{D}^{1/2}\|_2 = \frac{3}{2},$$

and

$$\frac{1}{2} = \|\mathbf{b}\mathbf{D}^{1/2}\|_2 - \frac{1}{2} \le |c_2|\cdot\|\mathbf{w}_1\mathbf{D}^{1/2}\|_2 \le \frac{1}{2} + \|\mathbf{b}\mathbf{D}^{1/2}\|_2 = \frac{3}{2}.$$

By the fact that $d \ge d_u \ge 1$ for any vertex $u \in H$ and that $\|\mathbf{w}_1\mathbf{D}^{1/2}\|_2^2 = \sum_u \mathbf{w}_1^2(u)d_u$, we have

$$d = \sum_u d\mathbf{w}_1^2(u) \ge \|\mathbf{w}_1\mathbf{D}^{1/2}\|_2^2 \ge \sum_u \mathbf{w}_1^2(u) = 1.$$

Furthermore, we have that

$$|c_{1,i}| \le \frac{3}{2},$$

and

$$\frac{1}{2d} \le |c_2| \le \frac{3}{2}.$$

Let $B_1 = \{u\colon\ |\mathbf{v}'_i(u) - c_{1,i}\cdot\mathbf{w}_1(u)|^2 \ge \frac{256C_1^2\beta_{\text{out}}}{\alpha\beta_{\text{in}}^2\mu_U}\}$, and $B_2 = \{u\colon\ |\mathbf{b}(u) - c_2\cdot\mathbf{w}_1(u)|^2 \ge \frac{256C_1^2\beta_{\text{out}}}{\alpha\beta_{\text{in}}^2\mu_U}\}$. By Inequality (11), the fact that $d_H(u) \ge 1$ for any $u \in V_H$ and an averaging argument, we have $|B_1| \le \frac{\alpha|U|}{2}$, and $|B_2| \le \frac{\alpha|U|}{2}$. Therefore, by letting $\widetilde{U} = U \setminus (B_1 \cup B_2)$, we have $|\widetilde{U}| \ge (1-\alpha)|U|$, and it holds that

$$|\mathbf{v}'_i(u) - c_{1,i}\cdot\mathbf{w}_1(u)|^2 \le \frac{256C_1^2\beta_{\text{out}}}{\alpha\beta_{\text{in}}^2\mu_U},$$

and

$$|\mathbf{b}(u) - c_2\cdot\mathbf{w}_1(u)|^2 \le \frac{256C_1^2\beta_{\text{out}}}{\alpha\beta_{\text{in}}^2\mu_U}.$$

for any $u \in \widetilde{U}$. Thus,

$$|\mathbf{v}'_i(u) - c_{1,i}\cdot\mathbf{w}_1(u)| \le \frac{16C_1}{\beta_{\text{in}}}\cdot\sqrt{\frac{\beta_{\text{out}}}{\alpha\cdot\mu_U}},$$

and

$$\left|\frac{c_{1,i}}{c_2}\mathbf{b}(u) - c_{1,i}\cdot\mathbf{w}_1(u)\right| \le \frac{|c_{1,i}|}{|c_2|}\frac{16C_1}{\beta_{\text{in}}}\cdot\sqrt{\frac{\beta_{\text{out}}}{\alpha\mu_U}}$$

$$\leq \frac{3/2}{1/(2d)} \frac{16C_1}{\beta_{\text{in}}} \cdot \sqrt{\frac{\beta_{\text{out}}}{\alpha\mu_U}}$$
$$\leq \frac{48dC_1}{\beta_{\text{in}}} \cdot \sqrt{\frac{\beta_{\text{out}}}{\alpha\mu_U}}.$$

Now for each $i \leq k$ we let $c_i := \frac{c_{1,i}}{c_2}$. Then $|c_i| \leq \frac{3/2}{1/(2d)} = 3d$, and for any vertex $u \in \widetilde{U}$, it holds that

$$|\mathbf{v}'_i(u) - c_i \cdot \mathbf{b}(u)| \leq |\mathbf{v}'_i(u) - c_{1,i} \cdot \mathbf{w}_1(u)| + \left| \frac{c_{1,i}}{c_2}\mathbf{b}(u) - c_{1,i} \cdot \mathbf{w}_1(u) \right|$$
$$\leq \frac{64dC_1}{\beta_{\text{in}}} \cdot \sqrt{\frac{\beta_{\text{out}}}{\alpha\mu_U}}.$$

Finally, by the definition of $\mathbf{b} = \frac{\mathbf{1}_{V_1,V_2}}{\sqrt{\mu_U}}$, we know that for each $i \leq k$,

- if $u \in V_1 \cap \widetilde{U}$, then $\left|\mathbf{v}'_i(u) - c_i \cdot \frac{1}{\sqrt{\mu_U}}\right| \leq \frac{64dC_1}{\beta_{\text{in}}} \cdot \sqrt{\frac{\beta_{\text{out}}}{\alpha\mu_U}}$,
- if $u \in V_2 \cap \widetilde{U}$, then $\left|\mathbf{v}'_i(u) + c_i \cdot \frac{1}{\sqrt{\mu_U}}\right| \leq \frac{64dC_1}{\beta_{\text{in}}} \cdot \sqrt{\frac{\beta_{\text{out}}}{\alpha\mu_U}}$. □

E.3. Proof of Lemma 9

Now we prove Lem. 9, which is restated in the following for the sake of readability.

Lemma 9. *Let $\alpha, \gamma \in (0,1)$. Let $G = (V, E, \sigma)$ be a signed, d-bounded degree, $(k, \beta_{in}, \beta_{out})$-clusterable graph. Let U be a subset of V such that $|U| \geq \gamma n$, $\beta^{inner}(G[U]) \geq \beta_{in}$ and $\beta_G(U) \leq \beta_{out}$. Then for all $t \geq \frac{4C_1^2(k+1)^6 \log n}{\beta_{in}^2}$, $\beta_{out} < \alpha_1 \beta_{in}^2$ where $\alpha_1 = \frac{\alpha\gamma}{800000C_1^2 kd^3}$, there exists a subset $\widetilde{U} \subseteq U$ with $\left|\widetilde{U}\right| \geq (1-\alpha)|U|$ and for all $u, v \in \widetilde{U}$, it holds that $\Delta_{uv} \leq \frac{1}{4nd}$.*

Proof. Since

$$\begin{aligned}
\mathbf{p}_u^t = \mathbf{1}_u \mathbf{W}^t &= \mathbf{1}_u \left(\frac{\mathbf{I} + \mathbf{D}^{-1}\mathbf{A}^\sigma}{2}\right)^t \\
&= \mathbf{1}_u \left(\frac{\mathbf{D}^{-1/2}(\mathbf{I} + \mathbf{D}^{-1/2}\mathbf{A}^\sigma\mathbf{D}^{-1/2})\mathbf{D}^{1/2}}{2}\right)^t \\
&= \mathbf{1}_u \mathbf{D}^{-1/2} \left(\frac{\mathbf{I} + \mathbf{D}^{-1/2}\mathbf{A}^\sigma\mathbf{D}^{-1/2}}{2}\right)^t \mathbf{D}^{1/2} \\
&= \mathbf{1}_u \mathbf{D}^{-1/2} \left(\mathbf{I} - \frac{\mathcal{L}_G}{2}\right)^t \mathbf{D}^{1/2} \\
&= \mathbf{1}_u \mathbf{D}^{-1/2} \sum_i (1 - \lambda_i/2)^t \mathbf{v}_i^\top \mathbf{v}_i \mathbf{D}^{1/2}.
\end{aligned}$$

Recall that $\mathbf{v}'_i = \mathbf{v}_i \mathbf{D}^{-1/2}$. We have that,

$$\begin{aligned}
\mathbf{p}_u^t \mathbf{D}^{-1/2} &= \mathbf{1}_u \mathbf{D}^{-1/2} \sum_i (1 - \lambda_i/2)^t \mathbf{v}_i^\top \mathbf{v}_i \\
&= \sum_i (1 - \lambda_i/2)^t \frac{\mathbf{v}_i(u)}{\sqrt{d_G(u)}} \mathbf{v}_i \\
&= \sum_i (1 - \lambda_i/2)^t \mathbf{v}'_i(u) \mathbf{v}_i.
\end{aligned}$$

Therefore we get that

$$\|\mathbf{p}_u^t \mathbf{D}^{-1/2} - \mathbf{p}_v^t \mathbf{D}^{-1/2}\|_2^2$$

$$
\begin{aligned}
&= \|\sum_{i=1}^{n}(\mathbf{v}_i'(u)-\mathbf{v}_i'(v))(1-\lambda_i/2)^t\mathbf{v}_i\|_2^2 \\
&= \sum_{i=1}^{n}(\mathbf{v}_i'(u)-\mathbf{v}_i'(v))^2(1-\lambda_i/2)^{2t} \\
&\le \sum_{i=1}^{k}(\mathbf{v}_i'(u)-\mathbf{v}_i'(v))^2 + \sum_{i=k+1}^{n}(\mathbf{v}_i'(u)-\mathbf{v}_i'(v))^2\,(1-\lambda_i/2)^{2t} \\
&\le \sum_{i=1}^{k}(\mathbf{v}_i'(u)-\mathbf{v}_i'(v))^2 + (1-\lambda_{k+1}/2)^{2t}\sum_{i=k+1}^{n}2\left(\frac{\mathbf{v}_i(u)^2}{d_G(u)}+\frac{\mathbf{v}_i(v)^2}{d_G(v)}\right) \\
&\le \sum_{i=1}^{k}(\mathbf{v}_i'(u)-\mathbf{v}_i'(v))^2 + 4\left(1-\frac{\beta_{\text{in}}^2}{2C_1^2(k+1)^6}\right)^{2t} \\
&\le \sum_{i=1}^{k}(\mathbf{v}_i'(u)-\mathbf{v}_i'(v))^2 + 4\exp\left(-\frac{\beta_{\text{in}}^2 t}{C_1^2(k+1)^6}\right),
\end{aligned}
$$

where in the fourth step we used that $(\mathbf{v}_i'(u)-\mathbf{v}_i'(v))^2 \le 2(\mathbf{v}_i'(u)^2+\mathbf{v}_i'(v)^2)$ by the Cauchy–Schwarz inequality and the definition of $\mathbf{v}_i'$. In the fifth step we used that $\sum_{i=k+1}^{n}\mathbf{v}_i(u)^2 \le \sum_{i=1}^{n}\mathbf{v}_i(u)^2 = 1$ and that $\lambda_{k+1} \ge \frac{\beta_{\text{in}}^2}{C_1^2(k+1)^6}$ by Lem. 5, as well as $d_G(u) \ge 1$, for any $u \in V$. Similarly,

$$
\|\mathbf{p}_u^t\mathbf{D}^{-1/2}+\mathbf{p}_v^t\mathbf{D}^{-1/2}\|_2^2 \le \sum_{i=1}^{k}(\mathbf{v}_i'(u)+\mathbf{v}_i'(v))^2 + 4\exp\left(-\frac{\beta_{\text{in}}^2 t}{C_1^2(k+1)^6}\right).
$$

We apply Lem. 6 on the set U and let $\widetilde{U} \subseteq U$, V_1, V_2 be the partition of U with the property specified in Lem. 6. Now we consider two vertices $u, v \in \widetilde{U}$. We now distinguish four cases:

- Case 1: If $u, v \in V_1$, then for all $i \in [k]$,

$$
\begin{aligned}
|\mathbf{v}_i'(u)-\mathbf{v}_i'(v)| &\le \left|\mathbf{v}_i'(u)-c_i\cdot\frac{1}{\sqrt{\mu_U}}\right| + \left|\mathbf{v}_i'(v)-c_i\cdot\frac{1}{\sqrt{\mu_U}}\right| \\
&\le \frac{128dC_1}{\beta_{\text{in}}}\cdot\sqrt{\frac{\beta_{\text{out}}}{\alpha\mu_U}}.
\end{aligned}
$$

  Thus,

$$
\begin{aligned}
\|\mathbf{p}_u^t\mathbf{D}^{-1/2}-\mathbf{p}_v^t\mathbf{D}^{-1/2}\|_2^2 &\le \sum_{i=1}^{k}(\mathbf{v}_i'(u)-\mathbf{v}_i'(v))^2 + 4\exp\left(-\frac{\beta_{\text{in}}^2 t}{C_1^2(k+1)^6}\right) \\
&\le \frac{20000kd^2C_1^2\beta_{\text{out}}}{\beta_{\text{in}}^2\cdot\alpha\mu_U} + 4\exp\left(-\frac{\beta_{\text{in}}^2 t}{C_1^2(k+1)^6}\right) \\
&\le \frac{20000kd^2C_1^2\beta_{\text{out}}}{\beta_{\text{in}}^2\cdot\alpha|U|} + 4\exp\left(-\frac{\beta_{\text{in}}^2 t}{C_1^2(k+1)^6}\right) \\
&\le \frac{20000kd^2C_1^2\beta_{\text{out}}}{\beta_{\text{in}}^2\cdot\alpha\gamma n} + 4\exp\left(-\frac{\beta_{\text{in}}^2 t}{C_1^2(k+1)^6}\right) \\
&\le \frac{1}{4nd},
\end{aligned}
$$

  where in the second to last inequality, we use the assumption that $\beta_{\text{out}} < \alpha_1\beta_{\text{in}}^2$ such that $\alpha_1 = \alpha_1(k,\alpha,\gamma,d) = \frac{\alpha\gamma}{800000C_1^2kd^3}$ and that $t \ge \frac{4C_1^2(k+1)^6\log n}{\beta_{\text{in}}^2}$.
- Case 2: If $u, v \in V_2$, then for all $i \in [k]$,

$$
|\mathbf{v}_i'(u)-\mathbf{v}_i'(v)| \le \left|\mathbf{v}_i'(u)+c_i\cdot\frac{1}{\sqrt{\mu_U}}\right| + \left|\mathbf{v}_i'(v)+c_i\cdot\frac{1}{\sqrt{\mu_U}}\right|
$$

$$\leq \frac{128dC_1}{\beta_{\text{in}}} \cdot \sqrt{\frac{\beta_{\text{out}}}{\alpha\mu_U}}.$$

Thus, similarly as above,

$$\begin{aligned}\|\mathbf{p}_u^t\mathbf{D}^{-1/2} - \mathbf{p}_v^t\mathbf{D}^{-1/2}\|_2^2 &\leq \frac{20000kd^2C_1^2\beta_{\text{out}}}{\beta_{\text{in}}^2 \cdot \alpha\mu_U} + 4\exp\left(-\frac{\beta_{\text{in}}^2 t}{C_1^2(k+1)^6}\right) \\ &\leq \frac{1}{4nd}.\end{aligned}$$

- Case 3: If $u \in V_1, v \in V_2$, then for all $i \in [k]$,

$$\begin{aligned}|\mathbf{v}_i'(u) + \mathbf{v}_i'(v)| &\leq \left|\mathbf{v}_i'(u) - c_i \cdot \frac{1}{\sqrt{\mu_U}}\right| + \left|\mathbf{v}_i'(v) + c_i \cdot \frac{1}{\sqrt{\mu_U}}\right| \\ &\leq \frac{128dC_1}{\beta_{\text{in}}} \cdot \sqrt{\frac{\beta_{\text{out}}}{\alpha\mu_U}}.\end{aligned}$$

Thus, similarly as above,

$$\begin{aligned}\|\mathbf{p}_u^t\mathbf{D}^{-1/2} + \mathbf{p}_v^t\mathbf{D}^{-1/2}\|_2^2 &\leq \sum_{i=1}^{k}(\mathbf{v}_i'(u) + \mathbf{v}_i'(v))^2 + 4\exp\left(-\frac{\beta_{\text{in}}^2 t}{C_1^2(k+1)^6}\right) \\ &\leq \frac{20000kd^2C_1^2\beta_{\text{out}}}{\beta_{\text{in}}^2 \cdot \alpha\mu_U} + 4\exp\left(-\frac{\beta_{\text{in}}^2 t}{C_1^2(k+1)^6}\right) \\ &\leq \frac{1}{4nd}.\end{aligned}$$

- Case 4: If $u \in V_2, v \in V_1$ then for all $i \in [k]$,

$$\begin{aligned}|\mathbf{v}_i'(u) + \mathbf{v}_i'(v)| &\leq \left|\mathbf{v}_i'(u) + c_i \cdot \frac{1}{\sqrt{\mu_U}}\right| + \left|\mathbf{v}_i'(v) - c_i \cdot \frac{1}{\sqrt{\mu_U}}\right| \\ &\leq \frac{128dC_1}{\beta_{\text{in}}} \cdot \sqrt{\frac{\beta_{\text{out}}}{\alpha\mu_U}}.\end{aligned}$$

Thus, similarly as above

$$\begin{aligned}\|\mathbf{p}_u^t\mathbf{D}^{-1/2} + \mathbf{p}_v^t\mathbf{D}^{-1/2}\|_2^2 &\leq \frac{20000kd^2C_1^2\beta_{\text{out}}}{\beta_{\text{in}}^2 \cdot \alpha\mu_U} + 4\exp\left(-\frac{\beta_{\text{in}}^2 t}{C_1^2(k+1)^6}\right) \\ &\leq \frac{1}{4nd}.\end{aligned}$$

Therefore, for any two vertices $u, v \in \widetilde{U}$, we have that

$$\Delta_{uv} = \min\{\|\mathbf{p}_u^t\mathbf{D}^{-1/2} - \mathbf{p}_v^t\mathbf{D}^{-1/2}\|_2^2, \|\mathbf{p}_u^t\mathbf{D}^{-1/2} + \mathbf{p}_v^t\mathbf{D}^{-1/2}\|_2^2\} \leq \frac{1}{4nd}.$$

□

E.4. Proof of Lemma 10

Now we prove Lem. 10, which is restated in the following for the sake of readability.

Lemma 10. *Let U_1 and U_2 be two disjoint subsets with $\beta_G(U_1), \beta_G(U_2) \leq \beta_{out}$. Let $0 < \alpha < 1$. For any $0 \leq t \leq \frac{\alpha}{8\beta_{out}}$, there exist subsets $\widehat{U_1} \subseteq U_1, \widehat{U_2} \subseteq U_2$ such that $|\widehat{U_1}| \geq (1-\alpha)|U_1|$, $|\widehat{U_2}| \geq (1-\alpha)|U_2|$, and for any $u \in \widehat{U_1}$ and $v \in \widehat{U_2}$, it holds that $\Delta_{uv} \geq \frac{1}{nd}$.*

Proof. Let $0 < \alpha < 1$. Consider a subset $C = (V_1, V_2)$ with $\beta_G(V_1, V_2) \leq \beta_{\text{out}}$. We first show that for any $t \geq 0$, there exists a subset $\widehat{C} \subseteq C$ such that $\text{vol}(\widehat{C}) \geq (1-\alpha)\,\text{vol}(C)$ and for any $v \in \widehat{C}$, $\sum_{w\in C} |\mathbf{1}_v \mathbf{W}^t(w)| \geq 1 - \frac{2t\beta_{\text{out}}}{\alpha}$. To do so, we first introduce some notations. For any vertex subset $C = (V_1, V_2) \subseteq V$, we define vectors y_{V_1,V_2} and $1_{V_1,V_2}$ as

$$\mathbf{y}_{V_1,V_2}(u) = \begin{cases} \frac{d_u}{\text{vol}(C)} & \text{if } u \in V_1, \\ -\frac{d_u}{\text{vol}(C)} & \text{if } u \in V_2, \\ 0 & \text{otherwise,} \end{cases}$$

$$\mathbf{1}_{V_1,V_2}(u) = \begin{cases} 1 & \text{if } u \in V_1, \\ -1 & \text{if } u \in V_2, \\ 0 & \text{otherwise.} \end{cases}$$

We first show the following result.

Claim 13. *For all $t \geq 0$, $\mathbf{y}_{V_1,V_2}\mathbf{W}^t\mathbf{1}_{V_1,V_2}^\top \geq 1 - t\beta_{out}$.*

Proof. We prove for any $t \geq 0$,

$$\mathbf{y}_{V_1,V_2}\mathbf{W}^t\mathbf{1}_{V_1,V_2}^\top - \mathbf{y}_{V_1,V_2}\mathbf{W}^{t+1}\mathbf{1}_{V_1,V_2}^\top \leq \beta_{\text{out}}.$$

Note that once the above inequality is proven, the claim follows from the fact that $\mathbf{y}_{V_1,V_2}\mathbf{W}^0\mathbf{1}_{V_1,V_2}^\top = \mathbf{y}_{V_1,V_2}\mathbf{1}_{V_1,V_2}^\top = 1$.

Let $\mathbf{x}$ be the vector such that $\mathbf{x}^\top = \mathbf{W}^t\mathbf{1}_{V_1,V_2}^\top$. Note that for any vertex $w \in V$, it holds that $|\mathbf{x}(w)| \leq 1$. Therefore,

$$\begin{aligned}
&\mathbf{y}_{V_1,V_2}\mathbf{W}^t\mathbf{1}_{V_1,V_2}^\top - \mathbf{y}_{V_1,V_2}\mathbf{W}^{t+1}\mathbf{1}_{V_1,V_2}^\top \\
&= \mathbf{y}_{V_1,V_2}(\mathbf{I} - \mathbf{W})W^t\mathbf{1}_{V_1,V_2}^\top \\
&= \mathbf{y}_{V_1,V_2}\mathbf{D}^{-1}\frac{\mathbf{D} - \mathbf{A}^\sigma}{2}\mathbf{W}^t\mathbf{1}_{V_1,V_2}^\top \\
&= \frac{1}{2\,\text{vol}(C)}\mathbf{1}_{V_1,V_2}(\mathbf{D} - \mathbf{A}^\sigma)\mathbf{W}^t\mathbf{1}_{V_1,V_2} \\
&= \frac{1}{2\,\text{vol}(C)}\sum_{(u,v)\in E}(\mathbf{1}_{V_1,V_2}(u) - \sigma(u,v)\mathbf{1}_{V_1,V_2}(v)) \cdot (\mathbf{x}(u) - \sigma(u,v)\mathbf{x}(v)) \\
&\leq \frac{1}{2\,\text{vol}(C)}\sum_{(u,v)\in E}|\mathbf{1}_{V_1,V_2}(u) - \sigma(u,v)\mathbf{1}_{V_1,V_2}(v)| \cdot |\mathbf{x}(u) - \sigma(u,v)\mathbf{x}(v)| \\
&\leq \frac{1}{2\,\text{vol}(C)}\sum_{(u,v)\in E} 2 \cdot |\mathbf{1}_{V_1,V_2}(u) - \sigma(u,v)\mathbf{1}_{V_1,V_2}(v)| \\
&= \frac{1}{\text{vol}(C)} \cdot (2|E_G^+(V_1,V_2)| + 2|E_G^-(V_1)| + 2|E_G^-(V_2)| + |E_G(C,\overline{C})|) \\
&\leq 2\beta_G(V_1,V_2) \leq 2\beta_{\text{out}}.
\end{aligned}$$

□

By the above claim, we have

$$\begin{aligned}
&\sum_{v\in C}\frac{d_v}{\text{vol}(C)}\sum_{w\in C}\left|\mathbf{1}_v\mathbf{W}^t(w)\right| \\
\geq &\sum_{v\in V_1}\frac{d_v}{\text{vol}(C)}\mathbf{1}_v\mathbf{W}^t\mathbf{1}_{V_1,V_2}^\top - \sum_{v\in V_2}\frac{d_v}{\text{vol}(C)}\mathbf{1}_v\mathbf{W}^t\mathbf{1}_{V_1,V_2}^\top \\
= &\mathbf{y}_{V_1,V_2}\mathbf{W}^t\mathbf{1}_{V_1,V_2}^\top \geq 1 - 2t\beta_{\text{out}}.
\end{aligned}$$

Thus,

$$\sum_{v\in C}\frac{d_v}{\text{vol}(C)}(1 - \sum_{w\in C}|\mathbf{1}_v\mathbf{W}^t(w)|) = 1 - \sum_{v\in C}\frac{d_v}{\text{vol}(C)}\sum_{w\in C}\left|\mathbf{1}_v\mathbf{W}^t(w)\right|$$

$$\leq 2t\beta_{\text{out}}.$$

Let $Q_C = \{v : \sum_{w\in C} |\mathbf{1}_v \mathbf{W}^t(w)| \leq 1 - \frac{2dt\beta_{\text{out}}}{\alpha}\}$. Then,

$$\begin{aligned}\sum_{v\in C} \frac{d_v}{\text{vol}(C)}(1 - \sum_{w\in C} \left|\mathbf{1}_v \mathbf{W}^t(w)\right|) &\geq \sum_{v\in Q_C} \frac{d_v}{\text{vol}(C)}(1 - \sum_{w\in C} \left|\mathbf{1}_v \mathbf{W}^t(w)\right|) \\ &\geq \frac{\text{vol}(Q_C)}{\text{vol}(C)} \frac{2dt\beta_{\text{out}}}{\alpha} \\ &\geq \frac{|Q_C|}{d|C|} \frac{2dt\beta_{\text{out}}}{\alpha} \ .\end{aligned}$$

Thus, $|Q_C| \leq \alpha|C|$. Therefore, if we set $\widehat{C} = C \setminus Q_C$, then $|\widehat{C}| \geq (1-\alpha)|C|$, and for any $v \in \widehat{C}$,

$$\sum_{w\in C} \left|\mathbf{1}_v \mathbf{W}^t(w)\right| \geq 1 - \frac{2dt\beta_{\text{out}}}{\alpha}.$$

Now for any two disjoint sets $C_1 = (V_1, V_2)$ and $C_2 = (V_1', V_2')$, we define $\widehat{C_1}$ and $\widehat{C_2}$ for C_1 and C_2, respectively. Thus, $|\widehat{C_1}| \geq (1-\alpha)|C_1|$ and $|\widehat{C_2}| \geq (1-\alpha)|C_2|$. Furthermore, for any $t \geq 1$ and $0 < \alpha < 1$, for any $u \in \widehat{C_1}$ and $v \in \widehat{C_2}$:

$$\sum_{w\in C_1} |\mathbf{1}_u \mathbf{W}^t(w)| \geq 1 - \frac{2dt\beta_{\text{out}}}{\alpha},$$

and

$$\sum_{w\in C_2} |\mathbf{1}_v \mathbf{W}^t(w)| \geq 1 - \frac{2dt\beta_{\text{out}}}{\alpha}.$$

Since C_1 and C_2 are disjoint, we have

$$\sum_{w\in C_1} |\mathbf{1}_v \mathbf{W}^t(w)| \leq 1 - \sum_{w\in C_2} |\mathbf{1}_v \mathbf{W}^t(w)| \leq \frac{2dt\beta_{\text{out}}}{\alpha},$$

and

$$\sum_{w\in C_2} |\mathbf{1}_u \mathbf{W}^t(w)| \leq 1 - \sum_{w\in C_1} |\mathbf{1}_u \mathbf{W}^t(w)| \leq \frac{2dt\beta_{\text{out}}}{\alpha}.$$

Let $\mathbf{q}_u^t$ be the vector such that $\mathbf{q}_u^t(w) = |\mathbf{1}_v \mathbf{W}^t(w)|$. Therefore, for any $t \geq 0$,

$$\begin{aligned}&\|(\mathbf{q}_u^t - \mathbf{q}_v^t)\mathbf{D}^{-1/2}\|_2^2 \\ &= \sum_{w\in V} (\mathbf{q}_u^t(w) - \mathbf{q}_v^t(w))^2 \frac{1}{d_G(w)} \\ &= \left(\sum_{w\in V} (\mathbf{q}_u^t(w) - \mathbf{q}_v^t(w))^2 \frac{1}{d_G(w)}\right) \cdot \left(\sum_{w\in V} d_G(w)) \cdot \frac{1}{\text{vol}(G)}\right) \\ &\geq \frac{(\sum_{w\in V} |\mathbf{q}_u^t(w) - \mathbf{q}_v^t(w)|)^2}{\text{vol}(G)} \qquad \text{(by Cauchy Schwarz inequality)} \\ &\geq \frac{(\sum_{w\in V} ||\mathbf{1}_u \mathbf{W}^t(w)| - |\mathbf{1}_v \mathbf{W}^t(w)||)^2}{\text{vol}(G)} \\ &\geq \frac{1}{\text{vol}(G)} \left[\sum_{w\in C_1} (|\mathbf{1}_u \mathbf{W}^t(w)| - |\mathbf{1}_v \mathbf{W}^t(w)|) + \sum_{w\in C_2} (|\mathbf{1}_v \mathbf{W}^t(w)| - |\mathbf{1}_u \mathbf{W}^t(w)|)\right]^2\end{aligned}$$

$$= \frac{1}{\mathrm{vol}(G)} \left[\sum_{w\in C_1} |\mathbf{1}_u \mathbf{W}^t(w)| - \sum_{w\in C_1} |\mathbf{1}_v \mathbf{W}^t(w)| + \sum_{w\in C_2} |\mathbf{1}_v \mathbf{W}^t(w)| - \sum_{w\in C_2} |\mathbf{1}_u \mathbf{W}^t(w)| \right]^2$$
$$\geq \frac{(2\cdot(1-\frac{2dt\beta_{\text{out}}}{\alpha}-\frac{2dt\beta_{\text{out}}}{\alpha}))^2}{\mathrm{vol}(G)}$$
$$= \frac{(2\cdot(1-\frac{4dt\beta_{\text{out}}}{\alpha}))^2}{\mathrm{vol}(G)}.$$

In particular, if $t \leq \frac{\alpha}{8d\beta_{\text{out}}}$, then $\|(\mathbf{q}_u^t - \mathbf{q}_v^t)\mathbf{D}^{-1/2}\|_2^2 \geq \frac{1}{\mathrm{vol}(G)} \geq \frac{1}{nd}$.

The lemma then follows from the fact that

$$\begin{aligned}\Delta_{uv} &= \min\{\|\mathbf{p}_u^t\mathbf{D}^{-1/2} - \mathbf{p}_v^t\mathbf{D}^{-1/2}\|_2^2, \|\mathbf{p}_u^t\mathbf{D}^{-1/2} + \mathbf{p}_v^t\mathbf{D}^{-1/2}\|_2^2\} \\ &\geq \|\mathbf{q}_u^t\mathbf{D}^{-1/2} - \mathbf{q}_v^t\mathbf{D}^{-1/2}\|_2^2.\end{aligned}$$

□

E.5. Proof of Lemma 11

Now we prove Lem. 11, which is restated in the following for the sake of readability.

Lemma 11. *Let $\alpha \in (0,1)$. Suppose $G = (V, E, \sigma)$ is a signed and $(k, \beta_{in}, \beta_{out})$-clusterable graph. Then there exists a set $V' \subseteq V$ of size $|V'| \geq (1-\alpha)|V|$ such that for all $u \in V'$ and all $t \geq \frac{C_1^2(k+1)^6 \log n}{\beta_{in}^2}$, we have that $\|\mathbf{p}_v^t\mathbf{D}^{-1/2}\|_2^2 \leq \frac{2k}{\alpha n}$.*

Proof. Recall that $\mathbf{v}_i$ is the i-th eigenvector of $\mathcal{L}_G^\sigma$, and $\mathbf{v}_i' = \mathbf{v}_i\mathbf{D}^{-1/2}$. For all $u \in V$, we set $\delta(u) = \sum_{i=1}^k \mathbf{v}_i'(u)^2$. Since we have that $\|\mathbf{v}_i\|_2^2 = 1$ and $d_G(u) \geq 1$,

$$\sum_{u\in V}\delta(u) = \sum_{u\in V}\sum_{i=1}^k \frac{\mathbf{v}_i(u)^2}{d_G(u)} = \sum_{i=1}^k\sum_{u\in V}\frac{\mathbf{v}_i(u)^2}{d_G(u)} \leq k.$$

Thus, the average value of $\delta(u)$ over all u is $\frac{k}{n}$. This implies that there exists a subset of vertices $V' \subseteq V$ of size $|V'| \geq (1-\alpha)|V|$ such that $\delta(u) \leq \frac{k}{\alpha n}$ for all $u \in V'$.

Furthermore, we have that $\mathbf{1}_u = \sum_{i=1}^n \mathbf{v}_i(u)\mathbf{v}_i$ and $\mathbf{p}_u^t\mathbf{D}^{-1/2} = \sum_{i=1}^n \mathbf{v}_i'(u)\left(1-\frac{\lambda_i}{2}\right)^t\mathbf{v}_i$. We now get that

$$\begin{aligned}\|\mathbf{p}_u^t\mathbf{D}^{-1/2}\|_2^2 &= \|\sum_{i=1}^n \mathbf{v}_i'(u)\left(1-\frac{\lambda_i}{2}\right)^t\mathbf{v}_i\|_2^2 \\ &= \sum_{i=1}^n \mathbf{v}_i'(u)^2\left(1-\frac{\lambda_i}{2}\right)^{2t} \\ &= \sum_{i=1}^k \mathbf{v}_i'(u)^2\left(1-\frac{\lambda_i}{2}\right)^{2t} + \sum_{i=k+1}^n \mathbf{v}_i'(u)^2\left(1-\frac{\lambda_i}{2}\right)^{2t} \\ &= \sum_{i=1}^k \mathbf{v}_i'(u)^2 + \left(1-\frac{\lambda_{k+1}}{2}\right)^{2t}\sum_{i=k+1}^n \mathbf{v}_i'(u)^2 \\ &\leq \delta(u) + \left(1-\frac{\lambda_{k+1}}{2}\right)^{2t} \\ &\leq \frac{k}{\alpha n} + \left(1-\frac{\beta_{\text{in}}^2}{2C_1^2(k+1)^6}\right)^{2t} \\ &\leq \frac{k}{\alpha n} + \exp\left(-\frac{\beta_{\text{in}}^2 t}{C_1^2(k+1)^6}\right)\end{aligned}$$

$$\leq \frac{2k}{\alpha n},$$

where we used that $\lambda_{k+1} \geq \frac{\beta_{\text{in}}^2}{C_1^2(k+1)^6}$ by Lem. 5 and in the last step we used that $t \geq \frac{C_1^2(k+1)^6 \log n}{\beta_{\text{in}}^2}$. □

E.6. Proof of Lemma 12

Now we prove Lem. 12, which is restated in the following for the sake of readability.

Lemma 12. *Let $\alpha \in (0,1)$ be a number such that $\frac{2k}{\alpha} \leq n$. Suppose $G = (V, E, \sigma)$ is a signed and $(k, \beta_{in}, \beta_{out})$-clusterable graph. Let $t \geq \frac{C_1^2(k+1)^6 \log n}{\beta_{in}^2}$. Let $V' \subseteq V$ be the set of vertices satisfying the property given by Lem. 11. Then* ESTDOTPROD*(u, v, t, α) outputs X_{uv} such that with probability $1 - 1/n^3$, it holds that*

$$\left| X_{uv} - \langle \mathbf{p}_v^t \mathbf{D}^{-1/2}, \mathbf{p}_u^t \mathbf{D}^{-1/2} \rangle \right| \leq \frac{1}{20nd}$$

for all $u, v \in V'$. Furthermore, ESTDOTPROD*(u, v, t, α) runs in time $O(\frac{d^2 k^{1.5} t \log n}{\alpha^{1.5}} \cdot \sqrt{n})$.*

Proof. This proof is based on Chiplunkar et al. (2018, Lem. 19). However, since the vectors we are analyzing may contain negative entries, we need to give a more refined analysis on the variance of the corresponding estimator.

Let $u, v \in V'$. Recall that $\mathbf{p}_u^t = \mathbf{1}_u \mathbf{W}^t$ and $\mathbf{p}_v^t = \mathbf{1}_v \mathbf{W}^t$. By Lem. 11 we have that $\|\mathbf{p}_u^t \mathbf{D}^{-1/2}\|_2, \|\mathbf{p}_v^t \mathbf{D}^{-1/2}\|_2 \leq \sqrt{\frac{2k}{\alpha n}}$. Let $\eta \in (0,1)$ be a parameter that will be specified later. Let R be an integer such that $R^2 \geq \frac{6}{\eta^2} \cdot \frac{2k}{\alpha n}$ and $R \geq \frac{24}{\eta^2} \cdot \left(\frac{2k}{\alpha n}\right)^{1.5}$.

For $x \in \{u, v\}$, we perform R lazy signed random walks from x of length t. Let $X_{x,w}^{r,s}$ be a random variable that is $\frac{1}{\sqrt{d_G(w)}}$ if the r'th walk that starts at vertex x ends at vertex w with sign $s \in \{+, -\}$. Set $X_{x,w}^r = X_{x,w}^{r,+} - X_{x,w}^{r,-}$. Observe that $\mathbf{E}\left[X_{x,w}^r\right] = \frac{\mathbf{p}_x^t(w)}{\sqrt{d_G(w)}}$ for all $w \in V$.

Let $\mathbf{m}_x^s(w)$ be the fraction of walks that start at x and end at w with sign s, for $x \in \{u, v\}$. Let $\mathbf{m}_x = \mathbf{m}_x^+ \mathbf{D}^{-1/2} - \mathbf{m}_x^- \mathbf{D}^{-1/2}$.

Now for any pair of vertices $u, v \in V'$, observe that

$$\langle \mathbf{m}_u, \mathbf{m}_v \rangle = \frac{1}{R^2} \sum_{w \in V} \left(\sum_{r_u=1}^{R} X_{u,w}^{r_u} \right) \left(\sum_{r_v=1}^{R} X_{v,w}^{r_v} \right).$$

This implies that

$$\begin{aligned}
\mathbf{E}\left[\langle \mathbf{m}_u, \mathbf{m}_v \rangle\right] &= \mathbf{E}\left[\frac{1}{R^2} \sum_{w \in V} \left(\sum_{r_u=1}^{R} X_{u,w}^{r_u} \right) \left(\sum_{r_v=1}^{R} X_{v,w}^{r_v} \right) \right] \\
&= \frac{1}{R^2} \sum_{w \in V} R \frac{\mathbf{p}_u^t(w)}{\sqrt{d_G(w)}} \cdot R \frac{\mathbf{p}_v^t(w)}{\sqrt{d_G(w)}} \\
&= \sum_{w \in V} \frac{\mathbf{p}_u^t(w)}{\sqrt{d_G(w)}} \frac{\mathbf{p}_v^t(w)}{\sqrt{d_G(w)}} = \langle \mathbf{p}_u^t \mathbf{D}^{-1/2}, \mathbf{p}_v^t \mathbf{D}^{-1/2} \rangle.
\end{aligned}$$

Next, we wish to compute $\mathbf{Var}\left[\langle \mathbf{m}_u, \mathbf{m}_v \rangle\right] = \mathbf{E}\left[\langle \mathbf{m}_u, \mathbf{m}_v \rangle^2\right] - \mathbf{E}\left[\langle \mathbf{m}_u, \mathbf{m}_v \rangle\right]^2$. We start by computing $\mathbf{E}\left[\langle \mathbf{m}_u, \mathbf{m}_v \rangle^2\right]$:

$$\begin{aligned}
\mathbf{E}\left[\langle \mathbf{m}_u, \mathbf{m}_v \rangle^2\right] &= \mathbf{E}\left[\frac{1}{R^4} \sum_{w \in V} \sum_{w' \in V} \sum_{r_u=1}^{R} \sum_{r'_u=1}^{R} \sum_{r_v=1}^{R} \sum_{r'_v=1}^{R} X_{u,w}^{r_u} X_{u,w'}^{r'_u} X_{v,w}^{r_v} X_{v,w'}^{r'_v} \right] \\
&= \frac{1}{R^4} \sum_{w \in V} \sum_{w' \in V} \sum_{r_u=1}^{R} \sum_{r'_u=1}^{R} \sum_{r_v=1}^{R} \sum_{r'_v=1}^{R} \mathbf{E}\left[X_{u,w}^{r_u} X_{u,w'}^{r'_u} X_{v,w}^{r_v} X_{v,w'}^{r'_v} \right].
\end{aligned}$$

We perform a case distinction in order to bound $\mathbf{E}\left[X_{u,w}^{r_u} X_{u,w'}^{r'_u} X_{v,w}^{r_v} X_{v,w'}^{r'_v} \right]$:

- If $w \neq w'$, then

$$\mathbf{E}\left[X_{u,w}^{r_u} X_{u,w'}^{r'_u} X_{v,w}^{r_v} X_{v,w'}^{r'_v}\right] = \begin{cases} \frac{\mathbf{p}_u^t(w)}{\sqrt{d_G(w)}} \cdot \frac{\mathbf{p}_v^t(w)}{\sqrt{d_G(w)}} \cdot \frac{\mathbf{p}_u^t(w')}{\sqrt{d_G(w')}} \cdot \frac{\mathbf{p}_v^t(w')}{\sqrt{d_G(w')}} & \text{if } r_u \neq r'_u \text{ and } r_v \neq r_{v'}, \\ 0 & \text{otherwise.} \end{cases}$$

- If $w = w'$, $r_u = r'_u$ and $r_v = r'_v$ then

$$\mathbf{E}\left[X_{u,w}^{r_u} X_{u,w}^{r_u} X_{v,w}^{r_v} X_{v,w}^{r_v}\right] \leq \frac{|\mathbf{p}_u^t(w)|}{\sqrt{d_G(w)}} \cdot \frac{|\mathbf{p}_v^t(w)|}{\sqrt{d_G(w)}} \cdot \frac{1}{\sqrt{d_G(w)}} \cdot \frac{1}{\sqrt{d_G(w)}}.$$

- If $w = w'$, $r_u = r'_u$ and $r_v \neq r'_v$ then

$$\mathbf{E}\left[X_{u,w}^{r_u} X_{u,w}^{r_u} X_{v,w}^{r_v} X_{v,w}^{r'_v}\right] \leq \frac{|\mathbf{p}_u^t(w)|}{\sqrt{d_G(w)}} \cdot \frac{|\mathbf{p}_v^t(w)|}{\sqrt{d_G(w)}} \cdot \frac{1}{\sqrt{d_G(w)}} \cdot \frac{|\mathbf{p}_v^t(w)|}{\sqrt{d_G(w)}}.$$

- If $w = w'$, $r_u \neq r'_u$ and $r_v = r'_v$ then

$$\mathbf{E}\left[X_{u,w}^{r_u} X_{u,w}^{r'_u} X_{v,w}^{r_v} X_{v,w}^{r_v}\right] \leq \frac{|\mathbf{p}_u^t(w)|}{\sqrt{d_G(w)}} \cdot \frac{|\mathbf{p}_v^t(w)|}{\sqrt{d_G(w)}} \cdot \frac{|\mathbf{p}_u^t(w)|}{\sqrt{d_G(w)}} \cdot \frac{1}{\sqrt{d_G(w)}}.$$

- If $w = w'$, $r_u \neq r'_u$ and $r_v \neq r'_v$ then

$$\mathbf{E}\left[X_{u,w}^{r_u} X_{u,w}^{r'_u} X_{v,w}^{r_v} X_{v,w}^{r'_v}\right] \leq \frac{|\mathbf{p}_u^t(w)|}{\sqrt{d_G(w)}} \cdot \frac{|\mathbf{p}_v^t(w)|}{\sqrt{d_G(w)}} \cdot \frac{|\mathbf{p}_u^t(w)|}{\sqrt{d_G(w)}} \cdot \frac{|\mathbf{p}_v^t(w)|}{\sqrt{d_G(w)}}.$$

Thus, we obtain that

$$\begin{aligned}
&\mathbf{E}\left[\langle \mathbf{m}_u, \mathbf{m}_v\rangle^2\right] \\
&= \frac{1}{R^4} \sum_{w\in V} \sum_{w'\in V} \sum_{r_u=1}^{R} \sum_{r'_u=1}^{R} \sum_{r_v=1}^{R} \sum_{r'_v=1}^{R} \mathbf{E}\left[X_{u,w}^{r_u} X_{u,w'}^{r'_u} X_{v,w}^{r_v} X_{v,w'}^{r'_v}\right] \\
&\leq \frac{R^2(R-1)^2}{R^4} \sum_{w\in V} \sum_{w'\neq w} \frac{\mathbf{p}_u^t(w)}{\sqrt{d_G(w)}} \cdot \frac{\mathbf{p}_v^t(w)}{\sqrt{d_G(w)}} \cdot \frac{\mathbf{p}_u^t(w')}{\sqrt{d_G(w')}} \cdot \frac{\mathbf{p}_v^t(w')}{\sqrt{d_G(w')}} \\
&\quad + \frac{1}{R^2} \sum_{w\in V} \frac{|\mathbf{p}_u^t(w)| \cdot |\mathbf{p}_v^t(w)|}{d_G^2(w)} + \frac{1}{R} \sum_{w\in V} \frac{|\mathbf{p}_u^t(w)| \cdot |\mathbf{p}_v^t(w)|^2}{d_G^2(w)} \\
&\quad + \frac{1}{R} \sum_{w\in V} \frac{|\mathbf{p}_u^t(w)|^2 \cdot |\mathbf{p}_v^t(w)|}{d_G^2(w)} + \sum_{w\in V} \frac{|\mathbf{p}_u^t(w)|^2 \cdot |\mathbf{p}_v^t(w)|^2}{d_G^2(w)} \\
&= \sum_{w,w'\in V} \frac{\mathbf{p}_u^t(w) \cdot \mathbf{p}_v^t(w) \cdot \mathbf{p}_u^t(w') \cdot \mathbf{p}_v^t(w')}{d_G(w) d_G(w')} \\
&\quad - \left(\frac{2R-1}{R^2}\right) \sum_{w\in V} \sum_{w'\neq w} \frac{\mathbf{p}_u^t(w) \cdot \mathbf{p}_v^t(w) \cdot \mathbf{p}_u^t(w') \cdot \mathbf{p}_v^t(w')}{d_G(w) d_G(w')} \\
&\quad + \frac{1}{R^2} \sum_{w\in V} \frac{|\mathbf{p}_u^t(w)| \cdot |\mathbf{p}_v^t(w)|}{d_G^2(w)} + \frac{1}{R} \sum_{w\in V} \frac{|\mathbf{p}_u^t(w)| \cdot |\mathbf{p}_v^t(w)|^2}{d_G^2(w)} \\
&\quad + \frac{1}{R} \sum_{w\in V} \frac{|\mathbf{p}_u^t(w)|^2 \cdot |\mathbf{p}_v^t(w)|}{d_G^2(w)},
\end{aligned}$$

This implies that

$$\mathbf{Var}\left[\langle \mathbf{m}_u, \mathbf{m}_v\rangle\right] = \mathbf{E}\left[\langle \mathbf{m}_u, \mathbf{m}_v\rangle^2\right] - \mathbf{E}\left[\langle \mathbf{m}_u, \mathbf{m}_v\rangle\right]^2$$

$$
\begin{aligned}
&\le \sum_{w,w'\in V} \frac{\mathbf{p}_u^t(w)\cdot\mathbf{p}_v^t(w)\cdot\mathbf{p}_u^t(w')\cdot\mathbf{p}_v^t(w')}{d_G(w)d_G(w')} \\
&\quad - \left(\frac{2R-1}{R^2}\right)\sum_{w\in V}\sum_{w'\neq w} \frac{\mathbf{p}_u^t(w)\cdot\mathbf{p}_v^t(w)\cdot\mathbf{p}_u^t(w')\cdot\mathbf{p}_v^t(w')}{d_G(w)d_G(w')} \\
&\quad + \frac{1}{R^2}\sum_{w\in V}\frac{|\mathbf{p}_u^t(w)|\cdot|\mathbf{p}_v^t(w)|}{d_G^2(w)} + \frac{1}{R}\sum_{w\in V}\frac{|\mathbf{p}_u^t(w)|\cdot|\mathbf{p}_v^t(w)|^2}{d_G^2(w)} \\
&\quad + \frac{1}{R}\sum_{w\in V}\frac{|\mathbf{p}_u^t(w)|^2\cdot|\mathbf{p}_v^t(w)|}{d_G^2(w)} - \left(\sum_{w\in V}\frac{\mathbf{p}_u^t(w)}{\sqrt{d_G(w)}}\frac{\mathbf{p}_v^t(w)}{\sqrt{d_G(w)}}\right)^2 \\
&\le \frac{1}{R^2}\sum_{w\in V}\frac{|\mathbf{p}_u^t(w)|}{\sqrt{d_G(w)}}\cdot\frac{|\mathbf{p}_v^t(w)|}{\sqrt{d_G(w)}} + \frac{1}{R}\sum_{w\in V}\frac{|\mathbf{p}_u^t(w)|}{\sqrt{d_G(w)}}\cdot\left(\frac{|\mathbf{p}_v^t(w)|}{\sqrt{d_G(w)}}\right)^2 \\
&\quad + \frac{1}{R}\sum_{w\in V}\left(\frac{|\mathbf{p}_u^t(w)|}{\sqrt{d_G(w)}}\right)^2\cdot\frac{|\mathbf{p}_v^t(w)|}{\sqrt{d_G(w)}} \\
&\quad + \frac{2}{R}\sum_{w,w'\in V}\frac{|\mathbf{p}_u^t(w)|\cdot|\mathbf{p}_v^t(w)|\cdot|\mathbf{p}_u^t(w')|\cdot|\mathbf{p}_v^t(w')|}{d_G(w)d_G(w')} \\
&\le \frac{1}{R^2}\|\mathbf{p}_u^t\mathbf{D}^{-1/2}\|_2\cdot\|\mathbf{p}_v^t\mathbf{D}^{-1/2}\|_2 + \frac{1}{R}\|\mathbf{p}_u^t\mathbf{D}^{-1/2}\|_2\cdot\|\mathbf{p}_v^t\mathbf{D}^{-1/2}\|_4^2 \\
&\quad + \frac{1}{R}\|\mathbf{p}_u^t\mathbf{D}^{-1/2}\|_4^2\cdot\|\mathbf{p}_v^t\mathbf{D}^{-1/2}\|_2 + \frac{2}{R}\left(\sum_{w\in V}\frac{|\mathbf{p}_u^t(w)|\cdot|\mathbf{p}_v^t(w)|}{d_G(w)}\right)^2 \\
&\le \frac{1}{R^2}\|\mathbf{p}_u^t\mathbf{D}^{-1/2}\|_2\cdot\|\mathbf{p}_v^t\mathbf{D}^{-1/2}\|_2 + \frac{1}{R}\|\mathbf{p}_u^t\mathbf{D}^{-1/2}\|_2\cdot\|\mathbf{p}_v^t\mathbf{D}^{-1/2}\|_2^2 \\
&\quad + \frac{1}{R}\|\mathbf{p}_u^t\mathbf{D}^{-1/2}\|_2^2\cdot\|\mathbf{p}_v^t\mathbf{D}^{-1/2}\|_2 + \frac{2}{R}\|\mathbf{p}_u^t\mathbf{D}^{-1/2}\|_2^2\cdot\|\mathbf{p}_v^t\mathbf{D}^{-1/2}\|_2^2,
\end{aligned}
$$

where in the last step we have used the Cauchy-Schwarz inequality and the monotonicity of the ℓ_p-norms[4], which gives $\|\mathbf{x}\|_4 \le \|\mathbf{x}\|_2$, for any vector $\mathbf{x}$.

Now recall that $\|\mathbf{p}_u^t\mathbf{D}^{-1/2}\|_2, \|\mathbf{p}_v^t\mathbf{D}^{-1/2}\|_2 \le \sqrt{\frac{2k}{\alpha n}}$. Recall that $\frac{2k}{\alpha} \le n$ and thus $\frac{2k}{\alpha n} \le 1$. This implies that

$$
\begin{aligned}
\mathbf{Var}\left[\langle \mathbf{m}_u, \mathbf{m}_v\rangle\right] &\le \frac{1}{R^2}\cdot\frac{2k}{\alpha n} + \frac{2}{R}\cdot\left(\frac{2k}{\alpha n}\right)^{1.5} + \frac{2}{R}\cdot\left(\frac{2k}{\alpha n}\right)^2 \\
&\le \frac{1}{R^2}\cdot\frac{2k}{\alpha n} + \frac{4}{R}\cdot\left(\frac{2k}{\alpha n}\right)^{1.5}.
\end{aligned}
$$

Now Chebyshev's Inequality implies that:

$$
\begin{aligned}
&\mathbf{Pr}\left(\left|\langle \mathbf{m}_u, \mathbf{m}_v\rangle - \langle \mathbf{p}_u^t\mathbf{D}^{-1/2}, \mathbf{p}_v^t\mathbf{D}^{-1/2}\rangle\right| \ge \eta\right) \\
&= \mathbf{Pr}\left(\left|\langle \mathbf{m}_u, \mathbf{m}_v\rangle - \mathbf{E}\left[\langle \mathbf{m}_u, \mathbf{m}_v\rangle\right]\right| \ge \eta\right) \\
&\le \frac{\mathbf{Var}\left[\langle \mathbf{m}_u, \mathbf{m}_v\rangle\right]}{\eta^2} \\
&\le \frac{1}{\eta^2}\cdot\left(\frac{1}{R^2}\cdot\frac{2k}{\alpha n} + \frac{4}{R}\cdot\left(\frac{2k}{\alpha n}\right)^{1.5}\right) \\
&\le \frac{1}{3}.
\end{aligned}
$$

[4]It is known that for $p, q \in (0, \infty)$ with $p \le q$, it holds that $\|\mathbf{x}\|_q \le \|\mathbf{x}\|_p$ for all vectors $\mathbf{x}$.

In the last inequality we have used that $R^2 \geq \frac{6}{\eta^2} \cdot \frac{2k}{\alpha n}$ and $R \geq \frac{24}{\eta^2} \cdot \left(\frac{2k}{\alpha n}\right)^{1.5}$.

Now we let $\eta = \frac{1}{20nd}$ and $R = \frac{40000 d^2 k^{1.5} \sqrt{n}}{\alpha^{1.5}}$ so that the above conditions on R are satisfied. Thus, with probability at least $1 - \frac{1}{3} = \frac{2}{3}$, the estimate $\langle \mathbf{m}_u, \mathbf{m}_v \rangle$ satisfies that

$$\left| \langle \mathbf{m}_u, \mathbf{m}_v \rangle - \langle \mathbf{p}_u^t \mathbf{D}^{-1/2}, \mathbf{p}_v^t \mathbf{D}^{-1/2} \rangle \right| \leq \frac{1}{20nd}.$$

Now note that the algorithm EstDotProd(u, v, t, α) repeatedly invokes the above subroutine for $h = O(\log n)$ times and outputs the median of the corresponding estimates $\langle \mathbf{m}_u, \mathbf{m}_v \rangle$, we are guaranteed that with probability at least $1 - 1/n^3$, the output X_{uv} satisfies that

$$\left| X_{uv} - \langle \mathbf{p}_u^t \mathbf{D}^{-1/2}, \mathbf{p}_v^t \mathbf{D}^{-1/2} \rangle \right| \leq \frac{1}{20nd}.$$

To obtain the runtime result, observe that for each run of the subroutine, the algorithm only performs R random walks of length t from both u and v, which can be done in $O(Rt)$ time. Thus, each of the vectors $\mathbf{m}_u$ and $\mathbf{m}_v$ has at most R non-zero entries and the dot product $\langle \mathbf{m}_u, \mathbf{m}_v \rangle$ can be computed in time $O(R)$. Finally, since we run the subroutine for $O(\log n)$ times, the total running time is thus $O(Rt \log n) = O(\frac{d^2 k^{1.5} t \sqrt{n} \log n}{\alpha^{1.5}})$.

This finishes the proof of the lemma. □

E.7. Proof of Thm. 4

Now we prove Thm. 4, which is restated in the following for the sake of readability.

Theorem 4. *Let $G = (V, E, \sigma)$ be a signed graph with $|V| = n$ vertices and maximum degree at most d. Suppose that G has a balanced $(k, \beta_{in}, \beta_{out})$-clustering $U_1, \cdots, U_k$, $\beta_{out} < \frac{\epsilon \beta_{in}^2}{C' \log(k) k^7 d^3 \log n}$, where C' is some sufficiently large constant, and $|U_i| \geq \gamma \frac{n}{k}$ for all $i = 1, \ldots, k$. There exists an algorithm that has query access to the adjacency list of G and constructs a clustering oracle in $O(\sqrt{n} \cdot \mathrm{poly}(\frac{kd \cdot \log n}{\epsilon \beta_{in}}))$ preprocessing time. Furthermore, with probability at least 0.9, the following hold:*

1. *Using the oracle, the algorithm can answer any* WhichCluster *query in $O(\sqrt{n} \cdot \mathrm{poly}(\frac{kd \cdot \log n}{\epsilon \beta_{in}}))$ time.*
2. *Let $P_i := \{u \in V : \text{WhichCluster}(u) = i\}$, $i \in [k]$, be the clusters defined by* WhichCluster. *Then there exists a permutation $\pi : [k] \to [k]$ such that for all $i \in [k]$, $\left| P_{\pi(i)} \triangle U_i \right| \leq O(\epsilon / \log k) |U_i|$.*

Proof. Given the above lemmas, we can prove our main theorem as follows. Recall that $C'' > 0$ is some large constant. Note that we have selected the random walk length $t = \frac{C'' k^6 d^3 \log n}{\beta_{\text{in}}^2}$.

Let $s = \frac{20k}{\gamma} \log(k)$, $\alpha = \frac{\epsilon}{90s}$. Recall that $\beta_{\text{out}} < \frac{\epsilon \beta_{\text{in}}^2}{C' \log(k) k^7 d^3 \log n}$. Note that $t \leq \frac{\alpha}{8 \beta_{\text{out}}}$ by changing appropriately large C''. Note that $\frac{2k}{\alpha} = \frac{1800 k^2 \log(k)}{\gamma \epsilon} \leq n$.

Correctness. Let $U_1, \ldots, U_k$ be a $(k, \beta_{\text{in}}, \beta_{\text{out}})$-clustering of G such that each cluster has size at least $\frac{\gamma n}{k}$, for some universal constant $\gamma > 0$. For any $u \in V$, let U_u be the cluster that contains u. We call a vertex u *bad*, if either:

- $u \in V \setminus V'$, where V' is the set as defined in Lem. 11 with $\alpha = \frac{\epsilon}{90s}$,
- $u \in U_u \setminus \widetilde{U_u}$, where $\widetilde{U_u}$ is as defined in Lem. 9 with $\alpha = \frac{\epsilon}{90s}$, or
- $u \in U_u \setminus \widehat{U_u}$, where $\widehat{U_u}$ is as defined in Lem. 10 with $\alpha = \frac{\epsilon}{90s}$.

Let B denote the set of all bad vertices. Note that

$$|B| \leq \left(\frac{\epsilon}{90s} + \frac{\epsilon}{90s} + \frac{\epsilon}{90s} \right) \cdot n = \frac{\epsilon \cdot n}{30s}.$$

We call a vertex u *good*, if it is not bad.

Note that since each cluster U has size at least $\frac{\gamma n}{k}$, it holds that

$$|B| \leq \frac{\gamma \epsilon \cdot n}{600k \log(k)} \leq O\left(\frac{\epsilon}{\log(k)} \right) |U|.$$

Thus, with probability at least $1 - \frac{\epsilon}{30s} \cdot s \geq 1 - \frac{1}{30}$, all the vertices in S are good. In the following, we will assume that this is the case.

Note further that since each cluster U satisfies that $|U| \geq \frac{\gamma n}{k}$ for some $\gamma = \Omega(1)$, it holds that with probability at least $1 - (1 - \frac{\gamma}{k})^s \geq 1 - \frac{1}{30k}$, there exists at least one vertex in S that is from cluster U. Thus, for all the k clusters U, with probability at least $1 - \frac{1}{30}$, there exists at least one vertex in S that is from cluster U.

By Lem. 11, we know that for any $v \in S$, $\|\mathbf{p}_v^t \mathbf{D}^{-1/2}\|_2^2 \leq \frac{2k}{n} \cdot \frac{90s}{\epsilon} = \frac{3600k^2 \log(k)}{\gamma\epsilon \cdot n}$. Let u, v be two different vertices in S. By Lem. 12, with probability at least $1 - \frac{1}{n^3}$, we can estimate each term $\langle \mathbf{p}_x^t \mathbf{D}^{-1/2}, \mathbf{p}_y^t \mathbf{D}^{-1/2} \rangle$ within an additive error at most $\frac{1}{20nd}$, for any $\{x, y\} \in \{u, v\}$. This also implies that with probability at least $1 - \frac{|S|^2}{n^3} \geq 1 - \frac{1}{n^2}$, for all vertex pairs $x, y \in S$, we have an estimate X_{xy} such that

$$\left| X_{xy} - \langle \mathbf{p}_x^t \mathbf{D}^{-1/2}, \mathbf{p}_y^t \mathbf{D}^{-1/2} \rangle \right| \leq \frac{1}{20nd}.$$

In the following, we will assume the above inequality holds for any $x, y \in S$.

Since v is good for each $v \in S$, we know that $X_{vv} \leq \frac{3600k^2 \log(k)}{\gamma\epsilon \cdot n} + \frac{1}{20nd} \leq \frac{4000k^2 \log(k)}{\gamma\epsilon \cdot n}$. Thus, Line 7 of Alg. 2 will not happen.

Note that

$$\Delta_{uv} = \min\{\|\mathbf{p}_u^t \mathbf{D}^{-1/2} - \mathbf{p}_v^t \mathbf{D}^{-1/2}\|_2^2, \|\mathbf{p}_u^t \mathbf{D}^{-1/2} + \mathbf{p}_v^t \mathbf{D}^{-1/2}\|_2^2\},$$

where

$$\|\mathbf{p}_u^t \mathbf{D}^{-1/2} - \mathbf{p}_v^t \mathbf{D}^{-1/2}\|_2^2 = \langle \mathbf{p}_u^t \mathbf{D}^{-1/2}, \mathbf{p}_u^t \mathbf{D}^{-1/2} \rangle - 2\langle \mathbf{p}_u^t \mathbf{D}^{-1/2}, \mathbf{p}_v^t \mathbf{D}^{-1/2} \rangle + \langle \mathbf{p}_v^t \mathbf{D}^{-1/2}, \mathbf{p}_v^t \mathbf{D}^{-1/2} \rangle,$$

and

$$\|\mathbf{p}_u^t \mathbf{D}^{-1/2} + \mathbf{p}_v^t \mathbf{D}^{-1/2}\|_2^2 = \langle \mathbf{p}_u^t \mathbf{D}^{-1/2}, \mathbf{p}_u^t \mathbf{D}^{-1/2} \rangle + 2\langle \mathbf{p}_u^t \mathbf{D}^{-1/2}, \mathbf{p}_v^t \mathbf{D}^{-1/2} \rangle + \langle \mathbf{p}_v^t \mathbf{D}^{-1/2}, \mathbf{p}_v^t \mathbf{D}^{-1/2} \rangle.$$

Since our estimates X_{vv}, X_{uv}, X_{uu} approximate $\langle \mathbf{p}_u^t \mathbf{D}^{-1/2}, \mathbf{p}_u^t \mathbf{D}^{-1/2} \rangle$, $\langle \mathbf{p}_u^t \mathbf{D}^{-1/2}, \mathbf{p}_v^t \mathbf{D}^{-1/2} \rangle$ $\langle \mathbf{p}_v^t \mathbf{D}^{-1/2}, \mathbf{p}_v^t \mathbf{D}^{-1/2} \rangle$ within an additive error $\frac{1}{20nd}$, respectively, we can approximate Δ_{uv} within an additive error at most $4 \cdot \frac{1}{20nd} = \frac{1}{5nd}$, i.e., the estimate δ_{uv} (at Line 13 of Alg. 2) satisfies that $|\delta_{uv} - \Delta_{uv}| \leq \frac{1}{5nd}$.

Now recall that each cluster U satisfies that $|U| \geq \frac{\gamma}{k} n$ for some $\gamma = \Omega(1)$ and that $\beta_{\text{out}} < \frac{\epsilon \beta_{\text{in}}^2}{C' \log(k) k^7 d^3 \log n}$. Note that the precondition of Lem. 9 is satisfied. Now let $S_U = S \cap U$, and let $u, v \in S$. Then:

- If u, v belong to the same cluster, by Lem. 9, we know that $\Delta_{uv} \leq \frac{1}{4nd}$. Then it holds that $\delta_{uv} \leq \Delta_{uv} + \frac{1}{5nd} < \frac{1}{2nd}$. Thus, an edge (u, v) will be added to H (at line 15 of Alg. 2).
- If u, v belong to two different clusters, by Lem. 10, we know $\Delta_{uv} \geq \frac{1}{nd}$. Then it holds that $\delta_{uv} \geq \Delta_{uv} - \frac{1}{5nd} > \frac{1}{2nd}$. Thus, an edge (u, v) will not be added to H.

Therefore, with probability at least $1 - \frac{1}{30} - \frac{1}{n^2} - \frac{1}{30} \geq 0.9$, the similarity graph H has the following properties:

1. all vertices in $V(H)$ (i.e., S) are good,
2. all vertices in S that belong to the same cluster U form a clique, denoted by H_U,
3. there is no edge between any two cliques H_{U_i} and H_{U_j} that correspond to two different clusters U_i, U_j,
4. there are exactly k cliques in H, each corresponding to some cluster.

Now let us consider a membership query, i.e., the subroutine WHICHCLUSTER(G, v, H, ℓ) for some vertex $v \in V$. We will show that any good vertex v will be correctly classified. In the following, we will assume that v is good.

Since all the vertices in S are good, we know that for any vertex $u \in U_v \cap S$, by Lem. 9, $\Delta_{uv} \leq \frac{1}{4nd}$, and by the same argument as above, with probability at least $1 - 1/n^2$, the estimate δ_{uv} (at Line 6 of Alg. 3) satisfies that $\delta_{uv} < \frac{1}{2nd}$. Thus, the label for v outputted by WHICHCLUSTER will be the same as the $\ell(u)$, the label of u.

On the other hand, for any other vertex $u \in S \setminus U_v$, by Lem. 10, $\Delta_{uv} \geq \frac{1}{nd}$, and by the same argument as above, with probability at least $1 - 1/n^2$, the estimate δ_{uv} (at Line 6 of Alg. 3) satisfies that $\delta_{uv} > \frac{1}{2nd}$. This further implies that the label of v will be different from the label of u.

Thus, all good vertices are correctly classified with probability at least $1 - \frac{2}{30} - \frac{n}{n^2} \geq 0.9$. Assuming this holds, then the set of misclassified vertices is a subset of all bad vertices, which implies that there exists a permutation $\pi : [k] \to [k]$ such that

$$|P_{\pi(i)} \triangle U_i| \leq |B| \leq O\left(\frac{\epsilon}{\log(k)}\right) \cdot |U_i|.$$

Running time. We first note that by Lem. 12, the subroutine EstDotProd(u, v, t, α) (i.e., Alg. 1) runs in time $O(\frac{d^2 k^{1.5} t \log n}{\alpha^{1.5}} \cdot \sqrt{n}) = O(\sqrt{n}\,\mathrm{poly}(\frac{kd \cdot \log n}{\epsilon \beta_{\text{in}}}))$.

For the algorithm BuildOracle, it invokes the subroutine EstDotProd for $O(s^2)$ times and uses the outputted estimates to construct the similarity graph H, which in total takes $O(s^2 \cdot \frac{d^2 k^{1.5} t \log n}{\alpha^{1.5}} \cdot \sqrt{n}) = O(\sqrt{n}\,\mathrm{poly}(\frac{kd \cdot \log n}{\epsilon \beta_{\text{in}}}))$ time.

For the algorithm WhichCluster, it invokes the subroutine EstDotProd for $O(s)$ times and uses uses the outputted estimates to answer, which in total takes $O(s \cdot \frac{d^2 k^{1.5} t \log n}{\alpha^{1.5}} \cdot \sqrt{n}) = O(\sqrt{n}\,\mathrm{poly}(\frac{kd \cdot \log n}{\epsilon \beta_{\text{in}}}))$ time. □

F. Implementation Details

We describe the practical implementations of our oracle data structures. We also discuss an *un*signed oracle and a heuristic algorithm for biclustering. Furthermore, we discuss how to practically determine the parameters for our algorithms.

Practical changes to our signed oracle. We start by giving some details on the implementation of our algorithm and the changes that we have made compared to the theoretical version.

First, we do not set δ_{uv} as described in Eqn. (3). Instead, we follow the intuition from Sec. D and use the vectors $\mathbf{r}_u^t$ rather than $\mathbf{p}_u^t$ in the definition of δ_{uv}. Recall that $\mathbf{r}_u^t = |\mathbf{p}_u^t|$, where the absolute values are taken component-wise. Therefore, we change Line 9 in Alg. 1 to $\mathbf{m}_x \leftarrow \left|(\mathbf{m}_x^+ - \mathbf{m}_x^-)\mathbf{D}^{-1/2}\right|$. Preliminary experiments (not reported here) showed that this provides slightly better results than when using the original choice of δ_{uv}. Furthermore, we only run the subroutine EstDotProd once (rather than $h = O(\log n)$ times).

Next, for a WhichCluster(v) query, the theoretical algorithm returns that v belongs to the cluster of vertex $u \in S$ if $\delta_{uv} \leq \frac{1}{2dn}$. However, in practice the the upper bound $\frac{1}{2dn}$ is not a suitable choice. Thus, we assign v to the cluster of $u = \arg\min_{w \in S} \delta_{wv}$.

Seeded and unseeded initialization. We consider two different initialization strategies: (1) when ground-truth seed nodes are available and (2) when use a randomized initialization.

In Case (1), when a small set of ground-truth seed nodes is available for each ground-truth cluster, we skip the preprocessing from Alg. 2 and take the vertex labels provided from the ground-truth seed nodes; we do not perform any other preprocessing.

In Case (2), we randomly sample a set S of vertices as in Alg. 2. However, we build the auxiliary graph H differently. Recall that in Alg. 2, we inserted all edges $(u, v) \in S \times S$ into H with $\delta_{uv} \leq \frac{1}{2dn}$. Preliminary experiments indicated that this upper bound is not a good choice in practice. Instead, we insert edges into H until it has k connected components (note that, initially, H has $|S|$ connected components). More concretely, we compute the pairwise distances δ_{uv} for all $u, v \in S$. Then we iterate over these distances in non-decreasing order and insert the corresponding edges into H until H has exactly k connected components. To obtain more robust distance estimates for δ_{uv}, we compute 5 samples of δ_{uv}, $u, v \in S$, and take the median; we only do this during the preprocessing phase for this algorithm (and *not* for queries as per Alg. 3).

Heuristic biclustering oracle. So far, we considered oracles for finding polarized communities $U_1, \ldots, U_k$ (see Def. 3). However, we did not consider partitioning each U_i into biclusters (V_{2i-1}, V_{2i}), that reveal the polarized groups in U_i. We now present a heuristic *biclustering oracle* for this purpose.

The heuristic biclustering oracle works exactly as the clustering oracle, with the following two changes. First, we do not take absolute values when computing $\mathbf{m}_x$, i.e., in Line 9 in Alg. 1 we set $\mathbf{m}_x \leftarrow (\mathbf{m}_x^+ - \mathbf{m}_x^-)\mathbf{D}^{-1/2}$. Second, when computing δ_{uv}, we now set $\delta_{uv} \leftarrow X_{uu} + X_{vv} - 2X_{uv}$. These changes correspond to our intuition from Sec. D that $\mathbf{m}_u$ approximates $\mathbf{p}_u^t \mathbf{D}^{-1/2}$ and that $\delta_{uv} \approx \|\mathbf{p}_u^t - \mathbf{p}_v^t\|_2^2$ is small *iff* u and v are from the same bicluster V_j. This intuition is also supported by our analysis via Lem.s 10 and 6. However, this is only a heuristic because it appears challenging to prove that our estimate δ_{uv} is large if $u \in V_1$ and $v \in V_2$; the main challenge is that we are only allowed a query time of $\widetilde{O}(\sqrt{n})$.

Unsigned oracle. To evaluate our oracles, it will be interesting to compare against an *unsigned oracle*, i.e., an oracle which ignores the edge signs and only considers the underlying unsigned graph. To this end, we consider unsigned versions of our clustering oracle and our heuristic biclustering oracle. Algorithmically, the only change is that we assume that all edges have sign $+$. The resulting unsigned oracle is almost identical to the oracle in (Czumaj et al., 2015).

Parameter tuning. To run our algorithms, one has to determine two crucial parameters: the length and the number of random walks. For both of them, our analysis requires the parameters α and β_{in}, which are not available in practice and it seems infeasible to estimate them. Therefore, we briefly describe how parameter tuning can be performed to obtain good choices for the length and the number of random walks.

Given a graph G, suppose that for a small set of vertices V_{labeled} we know their ground-truth communities. Now we build the oracle for several different parameters for the length and number of random walks. For each parameter setting, we run WHICHCLUSTER(v) for all $v \in V_{\text{labeled}}$ and check if v was classified correctly. At the end, we pick the parameter setting with the most correct answers.

Observe that the above procedure does not require a full clustering of G and can be used even when V_{labeled} is small. Further observe that we could also split V_{labeled} into a training set (used for the seed nodes), a validation set (used for determining the best parameters) and a test set (for estimating the overall accuracy).

G. Experiments on Synthetic Data

We evaluate our algorithms on synthetic datasets. We generated random graphs by starting with an empty graph and partitioning n vertices into equally-sized clusters $U_1, \dots, U_k$ with $U_i = (V_{2i-1}, V_{2i})$ for all $i \in [k]$. We inserted edges (u, v) with the following probabilities: p_{intra} if $u, v \in V_i$, p_{cross} if $u \in V_{2i-1}$ and $v \in V_{2i}$, q if $u \in U_i, v \notin U_i$. For all inserted edges, we set their sign to $+$ $(-)$ with probability p_{sign} if $u, v \in V_i$ $(u \in V_{2i-1}, v \in V_{2i})$ and to sign $-$ $(+)$ with probability $1 - p_{\text{sign}}$ otherwise. If $u \in U_i, v \notin U_i$ then we set the sign to $+$ with probability q_{sign} and to $-$ with probability $1 - q_{\text{sign}}$. When not stated otherwise, we set $n = 2000$, $k = 6$, $p_{\text{intra}} = 0.8$, $p_{\text{cross}} = 0.4$, $q = 0.05$, $p_{\text{sign}} = 0.8$ and $q_{\text{sign}} = 0.9$. For each experiment we have created 5 random graphs and we report average accuracies and their standard deviations.

Clustering experiments. We present our results for finding clusters $U_1, \dots, U_k$ in Fig. 3. We ran the oracles with $t = 2$ random walk steps and $R = 400$ random walks, unless stated otherwise. The seeded oracles and POLARSEEDS obtained 6 seed vertices from each U_i; the unseeded oracles randomly sampled $3k$ seeds.

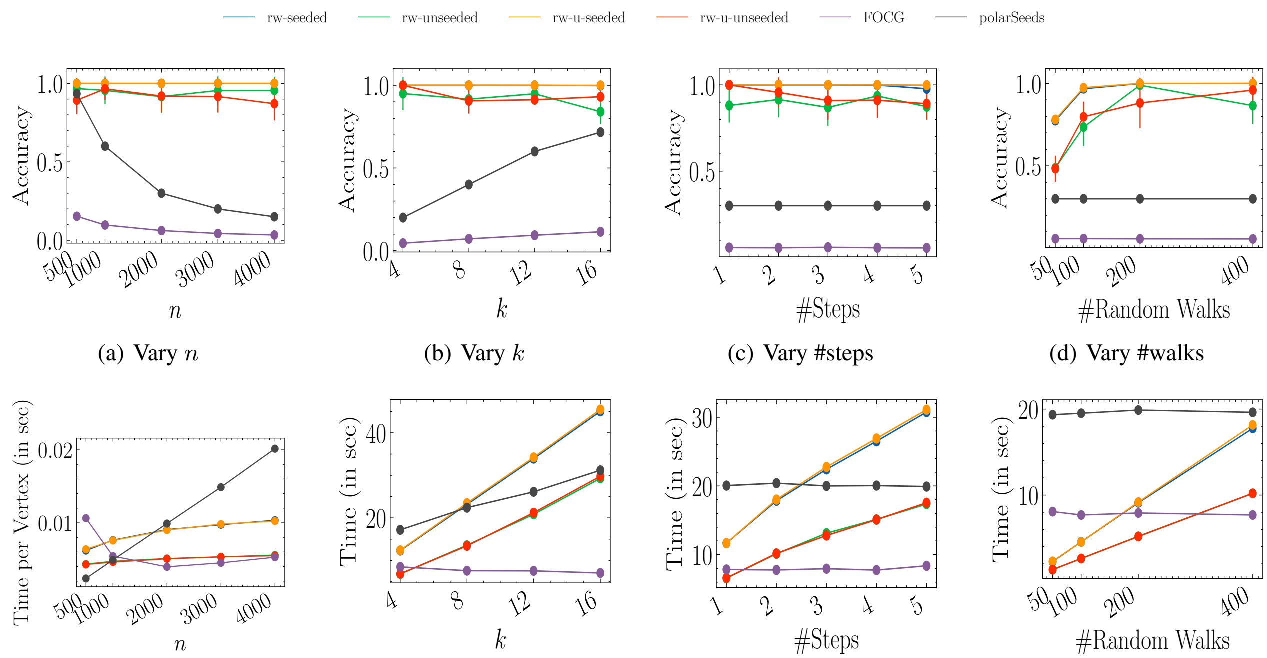
In Fig. 3(a) we vary the number of vertices n while keeping $k = 6$ fixed. RW-SEEDED and RW-U-SEEDED deliver almost perfect accuracy, i.e., they classify almost all vertices correctly; we note that in the plot, the lines of RW-SEEDED and RW-U-SEEDED are essentially identical and thus the line for RW-SEEDED is hard to see. RW-UNSEEDED and RW-U-UNSEEDED also deliver good results. Furthermore, POLARSEEDS works well when the clusters are small (for $n = 500$ there are 83 vertices in each cluster) but its performance decays as the clusters get larger. FOCG generally returns clusterings of low quality because it returns many clusters of very small sizes.

In Fig. 3(b), we fix n and vary $k = 4, 8, 12, 16$. Again, we observe a similar behavior as before: our oracles outperform the competitors, and the competitors improve for smaller clusters (k larger).

We also varied the parameters for the oracles. In Fig. 3(c), we set the number of random walk steps to $1, 2, 3, 4, 5$. We see that even with very short random walks, the algorithms deliver very good results. However, as the number of steps increases, the solution quality slightly decreases (see, e.g., RW-SEEDED or RW-U-UNSEEDED). This confirms the theoretical analysis of Lem.s 9 and 10.

In Fig. 3(d), we set the random walks lengths to $50, 100, 200, 4000$. RW-SEEDED and RW-U-SEEDED return excellent clusterings when at least $200 \approx 4.5 \cdot \sqrt{n}$ random walks are performed.

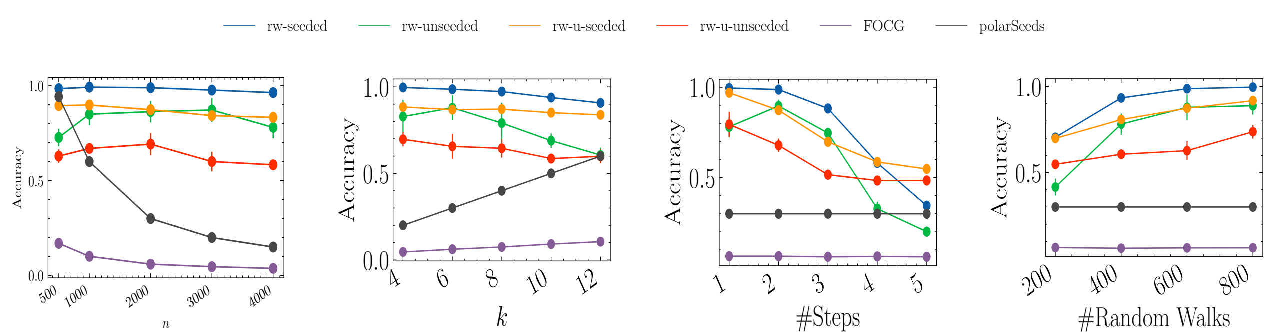
In Figs. 3(e)–(h) we report the running times of the algorithms. Our oracles scale linearly in the number of steps (Fig. 3(g)), and the number of random walks (Fig. 3(h)). Furthermore, since our number of seed nodes depends on the number of communities k, the oracles scale linearly in k (Fig. 3(f)). In Fig. 3(e) we report the running time, normalized by the number of vertices in the graph; for our oracle data structures this corresponds to the time they spend on each query. We observe that the query times of the oracles increases only very moderately as the number of vertices n increases; we blame this slight increase on the internal data structures (such as hash maps) that we use to store our graphs. This is in contrast to POLARSEEDS, for which the running time per vertex is increasing (Fig. 3(e)). For FOCG we observe that it scales linearly in the number of vertices (since in Fig. 3(e) the average time per vertex is nearly constant for $n \geq 1000$) and its running



(a) Vary n (b) Vary k (c) Vary #steps (d) Vary #walks

(e) Vary n: avg. query time (f) Vary k: time (g) Vary #steps: time (h) Vary #walks: time

Figure 3: Clustering results on synthetic data. We consider varying number of vertices n (Figs. (a) and (e)), varying number of communities k (Figs. (b) and (f)), varying random walk length (Figs. (c) and (g)), and varying number of random walks (Figs. (d) and (h)). Figs. (a)–(d) report the clustering accuracies, Fig. (e) reports the running time per vertex in seconds, and Figs. (f)–(h) report total running times in seconds. Markers are mean values and error bars are one standard deviation over 5 datasets.



(a) Vary n (b) Vary k (c) Vary #steps (d) Vary #walks

Figure 4: Biclustering results on synthetic data. We vary the number of vertices n (Fig. (a)), the number of communities k (Fig. (b)), the random walk length (Fig. (c)) and the number of random walks (Fig. (d)). We report the achieved accuracies; markers are mean values over 5 different datasets, and error bars are one standard deviation over the 5 datasets.

time slightly decreases as it finds better communities (Figs. 3(b) and 3(f))

Biclustering experiments. In Fig. 4 we present our results for finding biclusters $(V_1, V_2), (V_3, V_4), \dots, (V_{2k-1}, V_{2k})$. Thus, we run the biclustering versions of the algorithms. The oracles used $t = 2$ random walk steps and $R = 600$ random walks, unless stated otherwise. The seeded oracles and POLARSEEDS obtained 3 seed vertices from each V_i; the unseeded oracles randomly sampled $6k$ seed vertices in the preprocessing.

Again, our oracles obtain better results than the baseline algorithms, which typically return too small clusters. Furthermore, the signed oracles RW-SEEDED and RW-UNSEEDED outperform the unsigned oracles RW-U-SEEDED and RW-U-UNSEEDED, resp. This shows that the edge signs are necessary to split the clusters U_i into biclusters V_{2i-1} and V_{2i}. Compared to the clustering setting from before, the biclustering algorithms are more sensitive to the number steps (Fig. 4(c)), and they also require more random walks (Fig. 4(d)).

Conclusion. Our experiments suggest that our oracles outperform the baselines when the clusters are large. Also, to recover the biclusters (V_{2i-1}, V_{2i}), it is necessary to use the edge signs. Furthermore, the seeded methods outperform the unseeded methods.